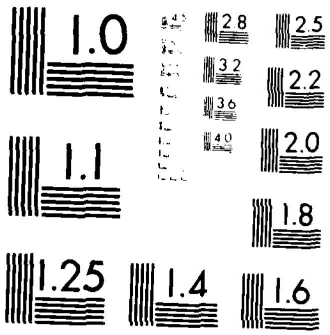
AD-A152 193 AN INVESTIGATION INTO THE CONTROL LIMITATIONS OF A BANK TO TURN MISSILE IN THE TERMINAL HOMING PHASE(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA B P ANDERSON SEP 84 1/3

UNCLASSIFIED F/G 17/7 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A152 193

NAVAL POSTGRADUATE SCHOOL

Monterey, California

DTIC
ELECTE
MAR 2 7 1985
D

E

THESIS

AN INVESTIGATION INTO THE CONTROL
LIMITATIONS OF A BANK TO TURN MISSILE
IN THE TERMINAL HOMING PHASE

by

Barton P. Anderson

September 1984

Thesis Advisor: M. D. Hewett

Approved for public release, distribution unlimited

85 03 11 119

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A152193	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Investigation Into The Control Limitations of a Bank to Turn Missile in the Terminal Homing Phase		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1984
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Barton Paul Anderson		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, CAlifornia 93943		12. REPORT DATE September 1984
		13. NUMBER OF PAGES 205
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Cruise Missile
Bank-to-turn homing
Proportional Navigation
Guidance and Control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The purpose of this thesis was to examine guidance and control deficiencies in a bank to turn (BTT) cruise missile with limited roll authority in the terminal homing phase of its mission. A six degree of freedom simulation of a typical BTT missile was translated into FORTRAN H from the Continuous System Modelling Program (CSMP) simulation language and run on the IBM System 370 computer. Tests were conducted with the revised

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

simulation program to examine the effects of electronic countermeasures (ECM) blinking and glint upon the missile's control system and accuracy against a simulated medium sized combatant vessel traveling at 20 knots perpendicular to the missile's track over the earth. In addition to the standard attack profile involving a popout attack, several other attack profiles were tested including skid-to-turn (STT) control laws and a ballistic trajectory. Miss distances varied from 3.7 feet without ECM or glint to 85 feet with ECM operating. Susceptibility of the missile to ECM blinking varied with the blinking frequency. The largest miss distances occurred with ECM frequencies below 0.2 Hz and near 6.0 Hz. Analysis of the data showed that errors at the low frequencies were primarily caused by the bank command loop of the autopilot. Those at the higher frequency were due to the roll rate command loop. Variation of the geometry of the missile's flight profile had no significant impact upon missile accuracy except that, without a popup maneuver, the roll rate channel demonstrated a marked decrease in effectiveness. Variation of the autopilot gain in the roll rate control loop changed the frequency at which degradation occurred but actually increased its effects. Skid to turn control laws were tested however the missile was unable to produce the necessary sideforce needed to track a passive target and produced undesireable coupling in the flight controls. An attempt to use the altitude command channel to fly a ballistic profile was unsuccessful due to instabilities created in the control system. It is recommended that a popup maneuver be included in the terminal guidance of a BTT cruise missile and that further tests be conducted to determine the extent to which autopilot midifications and gain adjustments can decrease the effectiveness of an ECM blinker against a BTT missile.

Approved for public release; distribution unlimited.

An Investigation into the Control Limitations of a Bank to Turn Missile in the Terminal Homing Phase

by

Barton P. Anderson
Commander, United States Navy
B.S., Wheaton College, 1970

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1984

Author: ______________________
Barton P. Anderson

Approved by: ______________________
M.D. Hewett, Thesis Advisor

____
M. Platzer, Chairman,
Department of Aeronautical Engineering

____
J.N. Dyer,
Department of Science and Engineering

ABSTRACT

The purpose of this thesis was to examine guidance and control deficiencies in a bank to turn (BTT) cruise missile with limited roll authority in the terminal homing phase of its mission. A six degree of freedom simulation of a typical BTT missile was translated into FORTRAN H from the Continuous System Modelling Program (CSMP) simulation language and run on the IBM System 370 computer. Tests were conducted with the revised simulation program to examine the effects of electronic countermeasures (ECM) blinking and glint upon the missile's control system and accuracy against a simulated medium sized combatant vessel traveling at 20 knots perpendicular to the missile's track over the earth. In addition to the standard attack profile involving a popout attack, several other attack profiles were tested including skid-to-turn (STT) control laws and a ballistic trajectory. Miss distances varied from 3.7 feet without ECM or glint to 85 feet with ECM operating. Susceptibility of the missile to ECM blinking varied with the blinking frequency. The largest miss distances occurred with ECM frequencies below 0.2 HZ and near 6.0 Hz. Analysis of the data showed that errors at the low frequencies were primarily caused by the bank command loop of the autopilot. Those at the higher frequency were due to the roll rate command loop. Variation of the geometry of the missile's flight profile had no significant impact upon missile accuracy except that, without a popup maneuver, the roll rate channel demonstrated a marked decrease in effectiveness. Variation of the autopilot gain in the roll rate control loop changed the frequency at which degradation occurred but actually increased its effects. Skid to turn control laws were tested

however the missile was unable to produce the necessary sideforce needed to track a passive target and produced undesireable coupling in the flight controls. An attempt to use the altitude command channel to fly a ballistic profile was unsuccessful due to instabilities created in the control system. It is recommended that a popup maneuver be included in the terminal guidance of a BTT cruise missile and that further tests be conducted to determine the extent to which autopilot modifications and gain adjustments can decrease the effectiveness of an ECM blinker against a BTT missile.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

The author wishes to acknowledge Dr. Marle D. Hewett, PhD, whose encouragement and advice contributed greatly to this project.

I. INTRODUCTION

A. BACKGROUND

Bank-To-Turn (BTT) control is utilized extensively on missiles which must cruise for long ranges within the atmosphere. These missiles utilize a primary lifting surface (wing) and smaller controlling surfaces as on a conventional airplane. This method has two primary advantages. First, the wing provides lift to support the missile's weight at a relatively high efficiency thereby permitting longer ranges for a given size engine and fuel load. Second, the lift vector can be positioned by banking the missile to provide large lateral accelerations resulting in excellent turn performance. Certain BTT cruise missile configurations, however, use differential tail for roll control as opposed to ailerons and suffer from poor roll rate and acceleration performance. It is the investigation into the control limitations of a BTT cruise missile configured this way in the terminal homing phase which is the subject of this thesis.

B. STATEMENT OF THE PROBLEM

In order to provide compact storage of a BTT missile, the main wings are usually folded back and designed to snap into position as the missile emerges from its cannister at launch. Because of this feature, it is generally not feasible to install roll control devices at the extremities of the wings. Roll control is normally provided by differential actuation of the tail fins of the missile. Because of their short moment arm and small area and because the main wing has a relatively large degree of roll damping, BTT missiles are limited in their ability to roll rapidly.

Because of the need to bank the missile in order to align its lift vector in the desired direction it has been suggested that the requirement for rapid roll maneuvering in the terminal phase of flight would limit the accuracy of the missile. In addition, natural fluctuations in the position of the radar target, known as glint, and artificial fluctuations due to the presence of electronic countermeasures (ECM) might further degrade the performance of a BTT missile.

C. MISSION SCENARIO

1. Control Configuration

The missile simulated in this thesis is a hypothetical bank to turn cruise missile with limited roll control authority. Its design incorporates characteristics typical of many similar designs. The missile is equipped with a standard rudder for yaw control and stabilators for both roll and pitch control. Inner loop closures for stabilization and command are included in the simulation. Command loop closures consist of normal acceleration, bank angle, and lateral acceleration. The lateral acceleration command system can be used as a turn coordinator in the bank-to-turn mode (normal) mode or as a lateral load factor (NY) command system in a skid-to-turn mode. Outer loop closures are provided for altitude and flight path angle. The autopilot control loop design is presented in detail in [Ref. 1].

2. Target

The target is assumed to be a surface combatant ship located initially 24,000 feet due North from the missile and moving East at a constant speed of 20 knots. It is assumed that the missile seeker tracks an aim point perfectly. The aim point is located nominally 10 feet above the ship's

waterline and amidships. This aim point continually shifts as a function of ECM blinking and a random glint simulation.

3. ECM Simulation

The ECM blinker simulation shifts the radar target seen by the missile's seeker forward and aft from the true target aim point by ±75 feet along the ships longitudinal axis at a specified frequency. The aim point is simultaneously shifted vertically ±10 feet at the same frequency.

4. Attack Profiles

The attack profile used as a baseline for this simulation began at 50 feet of altitude at a speed of Mach 0.75. The missile tracked toward the target using proportional navigation in azimuth and altitude hold at 50 feet. At a range of 18000 feet the missile rolled to 60 degrees of bank and turned away from the target to the right until the target line of sight was offset by 10 degrees. When the offset was reached, the missile climbed to an altitude of approximately 250 feet and then dove toward the target using proportional navigation in both azimuth and elevation. This mission profile is often referred to as a popout attack.

Variations of this mission included eliminating the 10 degree offset turn and/or the climb to altitude and substituting skid-to-turn control laws for some phase of the mission. A ballistic altitude profile was also attempted.

D. EXISTING WORK

In order to examine the existence of such problems and to test several proposed solutions, a six degree of freedom simulation of a typical BTT cruise missile was produced by LCDR Kent Watterson and published in [Ref. 1]. This simulation was produced using the IBM Continuous System Modelling

Program (CSMP III) simulation language. A detailed description of this language and its constructions is presented in [Ref. 2] and [Ref. 3]. The simulation included dynamics, autopilot, guidance and mission profiles. It did not represent any specific missile but, rather, included characteristics typical of missiles configured in this way. In order to overcome limitations imposed upon the simulation program by the available computer installation, this CSMP program was rewritten in extended FORTRAN H. This allowed greater flexibility and full utilization of the DISSPLA graphics programming package available at NPS. A complete copy of the program listing is presented in Appendix D.

E. SCOPE OF TESTS

The tests conducted with the revised simulation program were limited to examining the effects of ECM blinking and glint upon the missile's control system and accuracy against a simulated medium sized combatant vessel traveling at 20 knots perpendicular to the missile's track over the earth. Alternate attack profiles using modified flight geometry and, in some cases, skid-to-turn control laws were also tested. A listing of the different flight profiles examined is presented in table I.

For all flight tests of the missile, certain parameters were held constant. A list of these values is presented in table II.

TABLE I

Missile Attack Profile Test Configurations

	OFFSET TURN	POP-UP	KROLLP	TURN
BASELINE	X	X	0.5	BTT
II		X	0.5	BTT
III			0.5	BTT
IV		X	0.1	BTT
V		X	0.5	STT
VI		X	0.5	*

* 90 degree bank on ballistic terminal trajectory

TABLE II

Simulation Variables Held Constant

Variable Name	Value
Radar Burn-Through Range	500 ft
ECM Blinker Shifts:	
Longitudinal	± 75 ft
Lateral	± 00 ft
Vertical	± 10 ft
Baseline guidance scheme:	
Offset	10 deg
Popup Altitude	100 ft
Popup Range	13000 ft
Roll rate limit	75 dps

II. PROGRAM DESCRIPTION

A. INTEGRATION OF THE EQUATIONS OF MOTION

This simulation uses the linear, six degree of freedom equations of flight developed by Roskam in [Ref. 5:vol 1] and modified by Hewett in [Ref. 4]. The CSMP program developed by Patterson [Ref. 1] used a variable step Runge-Kutta integration method. The FORTRAN translation program uses a

INTEGRAL(YDOT DT) = Y + (YDOT)*DT (eqn 2.1)

simple Eulerian integration which is given by equation 2.1. The incremental time element, DT, is fixed at 0.01 seconds and the integration period lasts for less than 30 seconds.

B. PROGRAM NOMENCLATURE

A detailed description of the nomenclature used throughout the simulation program is presented in Appendix C. The variable names used in the FORTRAN translation are, with few exceptions, the same as those used in the the CSMP simulation.

C. AXIS SYSTEM

The simulation uses a right handed earth reference frame where the x-axis points North, the y-axis points East and the z-axis points down. However, altitude and vertical velocity are always given as positive upwards (i.e. ALTITUDE = -Z). For plotting the geographical track in the output routines, the axes are transformed so that the X,Y,and Z axes point East, North and upward, respectively.

D. PROGRAM ARCHITECTURE

The FORTRAN simulation program consists of an executive program which calls seven major subroutines which are briefly described as follows.

1. Executive Program

The main calling program is short and handles only three tasks. It increments the TIME variable for each integration cycle. It calls the output data storage routine, PREPAR, at the specified output interval and it controls the execution of multiple flights within a single program run changing one or more key variables between the runs.

2. Subroutine INIT

This subroutine contains a small section of executable statements which resets variables to their initial value when more than one flight is flown during a program run. Included with this subroutine is the BLOCK DATA subroutine which must be used to initialize all variables in named common areas. The majority of the BLOCK DATA subprogram is taken up with arrays listed in table form which contain the aerodynamic coefficient data for the missile. Static coefficients which are functions of one variable are shown in figures A.2 through A.9 Static coefficients which are functions of two parameters are presented in figures A.10 through A.18 Dynamic coefficients are assumed to be constant and are not presented graphically.

3. Subroutine MISSN

This subroutine dictates the mission profile. It is divided into sections which activate in sequence as the mission progresses. Each section takes the flight dynamics data for the missile, compares it with the target

acquisition data (generated in subroutine TGTNAV) and outputs vertical and horizontal acceleration commands in the geographic earth reference frame. These in turn are translated into commanded bank angle and normal load factor for the missile according to equations 2.2 and 2.3. A diagram

PHIC = ARCTAN(AYC/AZC) (eqn 2.2)

NZC = AZC cos(PHI) + AYC sin(PHI) (eqn 2.3)

of these vectors is given in figure A.1. Different terminal attack profiles are implemented using variations of this subroutine, MISSN1 and MISSN2, which are presented in Appendices E and F.

4. Subroutine APILOT

This subroutine takes the commanded normal load factor and bank angle and applies them to the missile autopilot system. A detailed discription of the design of the missile's autopilct is presented in reference [Ref. 1]. The output of the control system is delivered in terms of conventional airplane elevator, aileron and rudder control positions. These are mixed to obtain the commanded missile fin positions. The control limits of ±15 degrees are applied to the fins and these controls are then unmixed to obtain the limited conventional control positions. The dynamics of the servo actuators that move the tail surfaces are modelled as a first order real pole. Although CSMP-III provides macros that perform the simulation of many types of transfer functions within the control system only the first order real pole transfer function was necessary for this program. It is modelled in the FORTRAN translation using subroutine REALPL, presented in the program listing in Appendix C.

5. Subroutine AERO

Subroutine AERO uses two table lookup routines to retrieve the aerodynamic coefficients from the data presented in figures A.2 through A.18 Linear interpolations are used to obtain values between given parameters. Error messages are printed when the input parameters are outside the bounds of the data in the lookup table and these are suppressed after about 5 successive integration cycles. AERO completes the buildup process, uses these data to compute the forces and moments on the aircraft and then integrates the equations of motion to update all of the aircraft's flight parameters and position information.

6. Subroutine TGTNAV

The TGTNAV subroutine navigates the target vessel on a course of East at a steady speed of 20 knots. It shifts the position of the radar target relative to its real position according to the ECM and GLINT parameters. The GLINT offset is produced by multiplying the GLINT shift in each axis by a random number between -1 and 1. The GLINT offset is calculated every output interval rather than 100 times per second. The ECM offset is switched according to the sign of a sine wave which runs at the ECM blinking frequency, FREQ. These offsets are then added to the actual target position to produce the radar target position. Line of sight angles and rates are calculated from this information with the assumption that the seeker has perfect pointing capability.

7. Subroutine PREPAR

At intervals specified by the output counter, this subroutine is called and stores up to 20 variables in a large array call PTS. The output interval used for all tests

was 0.20 sec. The PTS array is passed to the output routines when the simulation run is completed. This subroutine also converts output variables from radian to degree format and, in the final attack phase, calculates four error functions. These error functions are time averaged differences between commanded variables (e.g. BANK or ROLL RATE) and their actual counterparts. These are later used to analyse the performance of the control system under various conditions.

8. <u>Subroutine OUTPUT</u>

OUTPUT produces 3 forms of output information. The primary data output lists the value of MISDST (the distance at which the missile passed the target at its closest approach), the value of the error functions at the end of the mission, and the ranges of all the variables stored. These data are also printed to another file followed by the full contents of the PTS array in tabular form. This gives a numerical history of all the output variables from the start to the finish of the mission. (Normally, to save disk space, this file was routed to a dummy variable. It was needed only when detailed data histories of a portion of the mission were required.)

OUTPUT also calls the necessary DISSPLA routines to print graphs of the output variables. The independent variable in six graphs is TIME. In the seventh graph the positions of the missile and the target ship are plotted in three dimensional space for each output interval. Each of the graphs in this subroutine are controlled by the setting of 7 flags in the first column of the data statement at the beginning of the routine (0 to pass over and 1 to plot).

III. BASELINE ATTACK CONFIGURATION

A. AUTOPILOT ROLL RATE COMMAND LOOP ADJUSTMENT

Initial testing of the simulation was conducted on the CSMP version of the program. The frequency of the ECM blinker was varied from 0.2 Hz to a maximum of 2.0 Hz and the roll performance of the missile was graphed. Figure A.19 shows the commanded roll rate and actual roll rate plotted against time for the duration of a thirty second flight straight toward the target at a constant altitude of 50 feet. The target's radar position was blinked at a rate of 0.4 Hz and roll rate command was limited to 75 degrees per sec. In the figure, the command oscillations increased in magnitude as the target range decreased and, after 24 seconds, the autopilot commanded the maximum rate with every shift of the target's apparent position. While the commanded roll rate remained at 75 degrees per second, the actual roll rate never exceeded 35 degrees per second. Figure A.20, which plots the fin positions as a function of time, shows that the fin servos never used more than 3 degrees (of the maximum 15) of travel in either direction. To remedy this problem, the missile autopilot roll rate command loop gain (KROLFT in the program) was increased from 0.1 to 0.5. The value of this gain had been set by Watterson [Ref. 1] using root locus based upon the perturbation equations of motion [Ref. 4] in steady state level flight. Figures A.21 and A.22 show the results of a subsequent run with the revised guidance loop. Steady state error in roll rate was significantly reduced and the full range of available flight controls (±15 deg.) was used. This difference in the autopilot was incorporated into the baseline program and remained throughout all subsequent tests.

E. BASELINE PROGRAM

In order to provide a baseline performance record against which to examine the effects of ECM and glint and/or alternate attack profiles on the accuracy of the missile and the performance of its control system, a standard, pop-out attack with an offset turn was selected and flown and is used as a standard for comparison. The parameters which apply to this baseline are listed in table II. Figures A.22 through A.28 are a complete record of the baseline program run without any ECM or glint offsets applied to the target. Figures A.29 through A.35 are a complete record of the baseline program run with the ECM blinker operating at 0.2 HZ and the glint feature operating. The complete tabular data output from this latter run is presented in Appendix B.

IV. FREQUENCY SCAN TESTS

A. ERROR FUNCTIONS

For testing the effects of glint and ECM at various blinking frequencies against the control system of the missile, a quantitative measure of the system's effectiveness was needed. Four error functions were developed for this purpose. The time weighted difference between the commanded value and the actual value of a variable was computed according to equation 4.1 This time weighted error was summed over all of the time intervals and divided by the

ERR = DT * ABS(COMMAND - VARIABLE) (eqn 4.1)

total time to produce the error function for the variable. The variables for which these functions were computed are

TABLE III

Error Function Variables

VARIABLE	COMMAND VARIABLE
1. BANK	BANK
2. ROLL RATE	ROLRT
3. AZIMUTH LOS RATE	0.0
4. ELEVATION LOS RATE	0.0

listed in Table III. In the terminal phase where proportional guidance is used in both the azimuth and elevation channels, the commanded azimuth and elevation angle rates are zero to produce a constant bearing intercept.

B. ECM PHASING

At low frequency blinking rates, the phase of the ECM blinker at the start of the mission had a very large effect on the miss distance. To minimize the distortion of the data due to this effect, a phase variable was added to the ECM generator to change the phase of the blinker at the start of each run. Four runs were conducted at each frequency using values of 0.0, PI/2, PI, (3/2)PI for the phase variable. The data for each frequency were averaged to get mean values for the miss distance and each error function.

C. BASELINE TEST RESULTS

1. ECM Frequency Scans

Four simulated flights were conducted at each frequency from 0.0 to 30 Hz. Glint was disabled for the course of these tests. The attack profile flown was the baseline popout attack mission. A graph of the mean value of the miss distance (MISDST) versus frequency is presented in figure A.36 The data show that maximum miss distance occurs in the very low frequency range of the the order of 0.2 Hz and again to a lesser degree in the vicinity of 6 Hz. Figures A.40 and A.44 are plots of the error function means against frequency for the autopilot command errors and the tracking errors respectively. These data show that the bank angle command loop is susceptible to ECM frequencies of the order of 0.2 Hz while the roll rate command loop is primarily responsible for the errors that occur at the higher frequencies in the range of 5 to 10 Hz. Figure A.44 also demonstrates that the time averaged tracking errors follow the same basic pattern.

Figures A.48 through A.53 demonstrate these effects in flight. Figures A.48 and A.51 show the bank angle and

roll rate performance of the baseline missile without ECM. Both variables track closely to their commanded values with the exception of a small, steady state error in the rate channel which is most evident at large commanded rates. Figures A.49 and A.50 show the effects of ECM at 0.4 and 6.0 Hz upon the bank channel. In figure A.49 significant errors exist in bank as the system cannot keep up with the large, sudden changes in commanded bank caused by the ECM shift of the target. The bracket in figure A.49 is drawn between two corresponding points to emphasize the large lag present in the channel. Roll rate tracks close to its commanded level at this frequency.

At 6.0 Hz, figures A.52 and A.53 show the opposite effect. In figure A.53 the bracket emphasizes the large lag that exists in the aircraft roll response to the rapid changes in rate command. The bank command loop at this frequency has effectively filtered out most of the high frequency input.

The results of the frequency scan tests showed that the baseline BTT cruise missile simulated by the program was more susceptible to ECM frequencies in the vicinity of 0.2 and 6.0 Hz due to the excitation of the bank and roll rate command loops respectively. If distances greater than 20 ft from the center of the target are considered likely misses, then the excitation of the roll rate command loop did not produce enough error to cause a likely miss. The best results, from the target's point of view, will be obtained with low blinking frequencies in the vicinity of 0.2 Hz.

2. Effects of Glint

In order to isolate the effects of glint, the baseline configuration was flown without ECM or glint and again with glint only. Figure A.33 shows a trace of the random glint

displacement applied to the target's position as a function of time. Figures A.23 through A.28, which trace the missile's load factor, bank angle, roll rate and flight controls without glint, may be compared with figures A.54 through A.57 which show the same traces for the mission with glint.

The miss distance recorded without glint and an ECM phase of 0 was 3.7 feet. The distance measured with glint was 9.4 feet. Although these distances are very small compared with the miss distances achieved with ECM, the degradation induced by glint was large (154 percent) compared to the best obtainable value. Ways of minimizing the effect of random perturbations in the target position due to radar glint will make a significant improvement in the missile's accuracy in the absence of ECM and should be developed.

Since the miss distances without ECM and glint were very small compared to those obtained with very slow blinking frequencies (0.05 to 0.2 Hz), further tests should be run concentrating on ECM in the very low frequency range. These tests should obtain a much larger sample of ECM phases in order to best define the shape of the miss distance curve below 0.2 Hz.

D. ALTERNATE CONFIGURATION FREQUENCY SCAN RESULTS

1. Mission Profile

Similar frequency scan profiles were flown using the MISSN1 (Appendix E) subroutine to generate the guidance commands for configurations II, III and IV. These attack profiles committed the offset turn and proceeded straight toward the target using proportional navigation in azimuth from start to finish. The popup maneuver was commenced at 15100 feet from the target. Of ranges from 20,000 to 5,000

feet which were tested, 15,000 feet produced the most consistent hits with a 200 foot popup altitude command. All subsequent tests of these missile attack configurations used 15,000 ft. popup range and a 200 ft. altitude command when the maneuver was performed.

An algorithm was added to the baseline proportional guidance scheme for the terminal phase which ensured that the missile rolled to place the nearest of the positive or negative Z-axis vectors on the direction commanded by the guidance system. This ensured that the missile would command negative load factor rather than trying to roll the missile upside down as it reached the apex of its climb. Azimuthal accelerations commanded by the guidance were still achieved by banking the missile except for configuration V.

A complete set of mission profile graphs for configurations II, III, and IV against a target with glint and ECM blinking at 0.2 Hz are presented in figures A.58 through A.75

2. Frequency Scan Results

a. Miss Distances

Each configuration was flown against the target four times per test frequency. The tests covered a range of blinker frequencies from 0.05 through 30.0 Hz. The mean miss distances recorded are graphically presented as a function of frequency in figures A.37 through A.39 The results obtained were very similar to those obtained from the baseline configuration. There were two areas of higher than normal errors, one at low frequency below 0.2 Hz and another at a higher frequency near 6.0 Hz. Table IV compares the miss distances for each of the configurations.

The maximum values that occurred for all configurations appeared at the same frequencies with one

TABLE IV

Maximum Miss Distances

CONFIGURATION	FREQ. RANGE (HZ)	LOCATION (HZ)	MAGNITUDE
BASELINE	0.20 - 20.0	≤ 0.20 6.00	≥ 45 22
II	0.05 - 21.0	≤ 0.05 6.00	≥ 75 17
III	0.10 - 30.0	≤ 0.10 5.50	≥ 75 17
IV	0.10 - 30.0	≤ 0.10 N/A	≥ 75 N/A

exception: changing the roll rate gain from 0.5 to 0.[illegible] eliminated the maximum at the higher frequency. In addition, the magnitude of the errors did not differ significantly. (The baseline shows a smaller magnitude because the data do not extend below 0.2 Hz. while the other configurations were tested down to 0.1 and 0.05 Hz). Changing the attack geometry of the missile did not significantly alter its susceptibility to ECM jamming within the scope of these tests. Altering the gain of the roll rate command channel in the missile autopilot significantly decreased its susceptibility to ECM blinking at higher frequencies. Further testing should be conducted to determine the extent to which autopilot modifications and gain adjustments can decrease the effectiveness of an ECM blinker against a bank to turn missile.

1. Autopilot Errors

Figures A.40 through A.43 graphically present the error functions for both the bank angle and roll rate command loops within the autopilot. These functions are representative of the ability of the missile to follow the

commands given it by the autopilot (the higher the function, the poorer the performance). As with the baseline configuration these figures demonstrate that the bank angle loop contributed most to the errors at low frequency and the roll rate loop contributed most at the higher frequency. Table V

TABLE V

Autopilot Errors

CONFIGURATION	BANK ERROR		RATE ERROR	
	FREQ. (HZ)	MAGNITUDE	FREQ. (HZ)	MAGNITUDE
BASELINE	0.4	0.22	7.0	0.19
II	0.6	0.17	8.0	0.18
III	0.5	0.18	8.0	0.27
IV	0.6	0.21	2.0	0.37

is a summary of these graphs.

Magnitude of the bank error function and the frequency at which it occurred were not significantly altered in any one of the tested configurations. Changing the geometry of the attack had no effect on the frequency at which ECM was most effective against the roll rate control system, however the magnitude of the errors were increased by approximately 50 percent when the popup maneuver was eliminated (configuration III).

Decreasing the roll rate autopilot gain from 0.5 to 0.1 (configuration IV) moved the resonant frequency for the roll rate command system to a lower frequency but

increased the magnitude of the errors by more than 100 percent. This effect is reflected in the miss distance graphs (figures A.36 through A.39) in the disappearance of the distinct maximum at 6 HZ and a widening of the lower maximum (figure A.39). Altering the autopilot gain was effective at moving the resonant frequency to a different region but could not eliminate its effect.

c. Tracking System Errors

Errors in the tracking loops are charted in figures A.44 through A.47. These errors follow the trends of the autopilot and miss distance errors. At the lower frequencies, azimuth performance was dominant while at higher frequencies the elevation tracking loop experienced the largest degradation.

3. Skid To Turn Guidance Results

The MISSN1 subroutine was further modified to allow the lateral load factor command variable, NYC, to be set according to guidance commands rather than being kept at zero for turn coordination purposes. The commanded bank angle was set to zero in the terminal phase in order to examine the effectiveness of lateral G command. No changes to the basic dynamics of the autopilot were made. The missile was flown in this configuration against a passive target. Figures A.76 through A.81 present the full data set from this test. The missile splashed into the water 99 feet left and short of the target. Once the missile came within 5 seconds of impact, cross coupling between the rudder channel and normal load factor, roll rate and bank can be seen in the figures. Although the rudder commands were never saturated, neither could the lateral load factor control loop create enough sideforce to follow the ship's lateral drift to the right. The addition of ECM and/or glint would have

only worsened the performance of the missile in this configuration. No further tests of this configuration were conducted. The use of skid-to-turn control laws could not produce sufficient sideforce to adequately follow a passive crossing target and produced excessive coupling into the longitudinal and lateral flight controls of the missile.

4. Ballistic Trajectory

Because the majority of the apparent target shift with ECM blinking occurs in the horizontal plane, an attempt was made to place the missile on a ballistic trajectory and then roll the aircraft to 90 degrees angle of bank until impact using the primary load factor to follow the ECM target and lateral load factor to maintain the ballistic trajectory. In order to fly the ballistic trajectory, the altitude hold system was driven by a commanded altitude slaved to a parabolic trajectory derived from the missile's vertical speed and range to the target according to equation

$$ALT = HMDOT*RANGE/VH + (G/2)*(RANGE/VT)^2 + 10 \quad \text{(eqn 4.2)}$$

4.2. where HMDOT, VH and VT are the vertical, horizontal and total speeds of the missile. The controlling subroutine used for this mission was MISSN2 and is presented in Appendix F.

Figures A.82 through A.85 show that the addition of the dynamics of the altitude command loop made the missile's control system unstable. Oscillations to the limits occurred in normal load factor and in roll rate. Considerable cross coupling occurred between the lateral-directional and longitudinal dynamics of the missile. The attempt to fly a ballistic trajectory using the existing altitude control system was unsuccessful. In order to fly the attempted profile, a major redesign of the missile's autopilot would be necessary.

V. CONCLUSIONS

The conclusions listed below were derived from analysis of the results of simulated flights conducted using the baseline popout attack profile configuration, three variations of the baseline attack, a skid-to-turn control configuration and a ballistic altitude trajectory.

A. BASELINE CONFIGURATION TESTS

At low frequency blinking rates, the phase of the ECM blinker had a very large effect on the miss distance.

The best obtainable performance for the baseline mission without ECM or glint was a miss distance of 3.7 feet. The addition of GLINT produced a miss distance of 9.7 feet, a degradation of 154 percent.

The bank angle command loop of the missile autopilot in the baseline configuration was especially susceptible to ECM frequencies of the order of 0.2 Hz while the roll rate command loop was primarily affected at the higher frequencies in the range of 5 to 10 Hz. The time averaged tracking errors also followed the same basic pattern.

If distances greater than 20 ft from the center of the target are considered likely misses, then the excitation of the roll rate command loop did not produce enough error to cause a likely miss. The best results, from the target's point of view, will be obtained with low blinking frequencies in the vicinity of 0.2 Hz.

B. ALTERNATE ATTACK PROFILE CONFIGURATIONS

In terms of the average miss distances measured, changing the flight geometry of the missile did not signifi-

cantly alter its susceptibility to ECM jamming within the scope of these tests.

Altering the gain of the roll rate command channel in the baseline missile autopilot significantly decreased its susceptibility to ECM blinking at higher frequencies.

Changing the geometry of the attack had no effect on the magnitude of the bank error function and the frequency at which its maximum occurred.

Changing the roll rate gain from 0.5 to 0.1 had no noticeable affect on the magnitude of the bank error function and the frequency at which its maximum occurred.

Changing the geometry of the attack had no effect on the frequency at which ECM was most effective against the roll rate control system, however the magnitude of the errors were increased by approximately 50 percent when the popup maneuver was eliminated (configuration III).

Decreasing the roll rate autopilot gain from 0.5 to 0.1 (configuration IV) moved the resonant frequency for the roll rate command system to a lower frequency but increased the magnitude of the errors by more than 100 percent. This effect was reflected in the miss distance data by the disappearance of the distinct maximum at 6 HZ and a widening of the lower maximum. Altering the autopilot gain was effective at moving the resonant frequency to a different region but could not eliminate its effect and, in this case enlarged it.

Errors in the azimuth and elevation tracking loops closely followed the trends of the autopilot and miss distance errors. At the lower frequencies, azimuth performance was dominant while at higher frequencies the elevation tracking loop experienced the largest degradation.

C. SKID TO TURN CONTROL

The use of skid-to-turn control laws could not produce sufficient sideforce to adequately follow a passive crossing target and produced excessive coupling into the longitudinal and lateral flight controls of the missile.

D. BALLISTIC ATTACK PROFILE

The attempt to fly a ballistic trajectory using the existing altitude control system was unsuccessful. In order to fly the attempted profile, a major redesign of the missile's autopilot would be necessary.

VI. RECOMMENDATIONS

Ways of minimizing the effect of random perturbations in the target position due to radar glint will make a significant improvement in the missile's accuracy in the absence of ECM and should be developed.

Further testing should be conducted to determine the extent to which autopilot modifications and gain adjustments can decrease the effectiveness of an ECM blinker against a bank to turn missile.

Since the elimination of a popup increased roll rate errors by 50 percent, a popup profile is recommended for the terminal phase of a BTT cruise missile. Further testing should be conducted to determine the effects of different popup profiles on the susceptibility of the roll rate command system to ECM blinking.

Since the miss distances without ECM and glint were very small compared to those with very slow blinking frequencies (0.05 to 0.2 Hz), further tests should be run concentrating on ECM in the very low frequency range. These tests should obtain a much larger sample of ECM phases in order to best define the shape of the miss distance curve below 0.2 Hz.

APPENDIX A
FIGURES

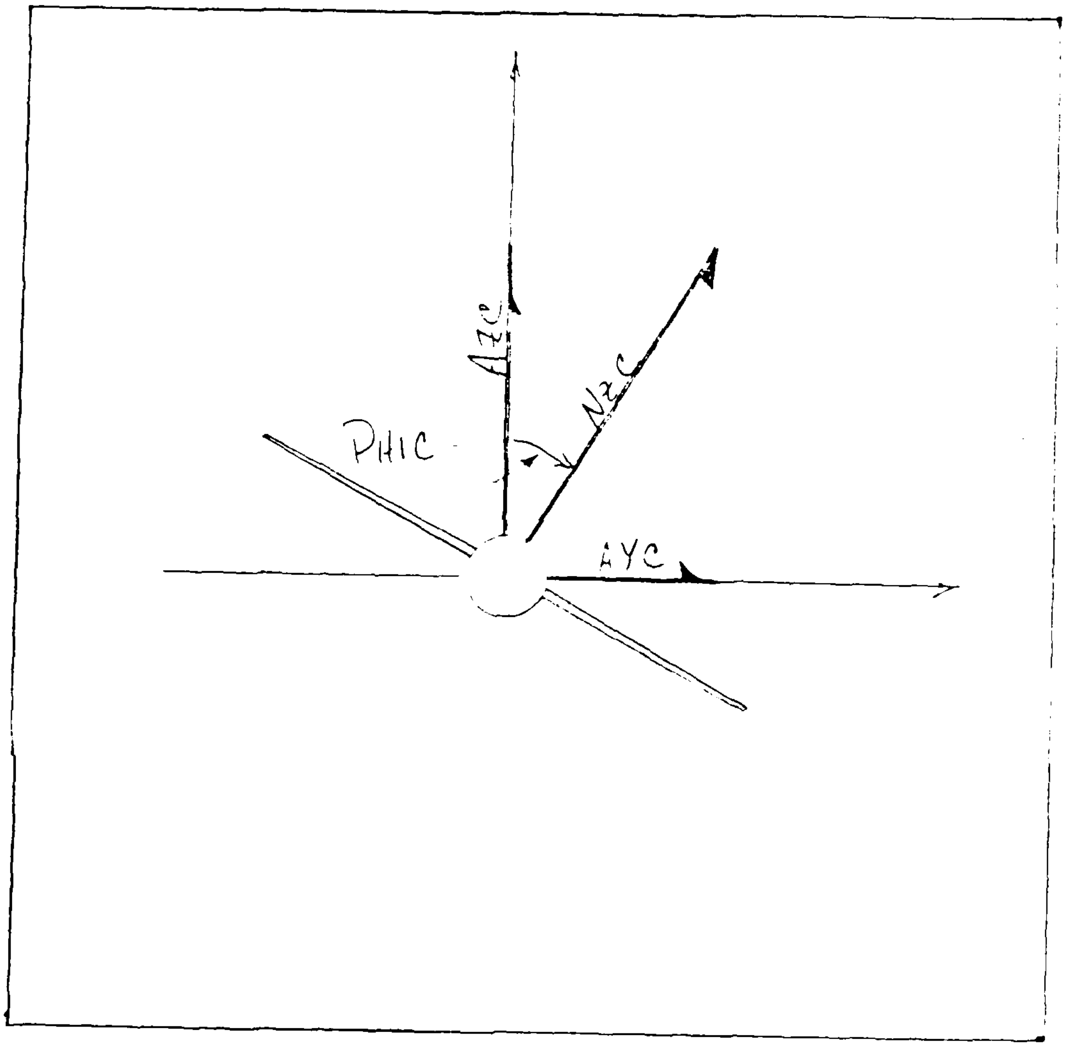


Figure A.1 Load Factor Commands.

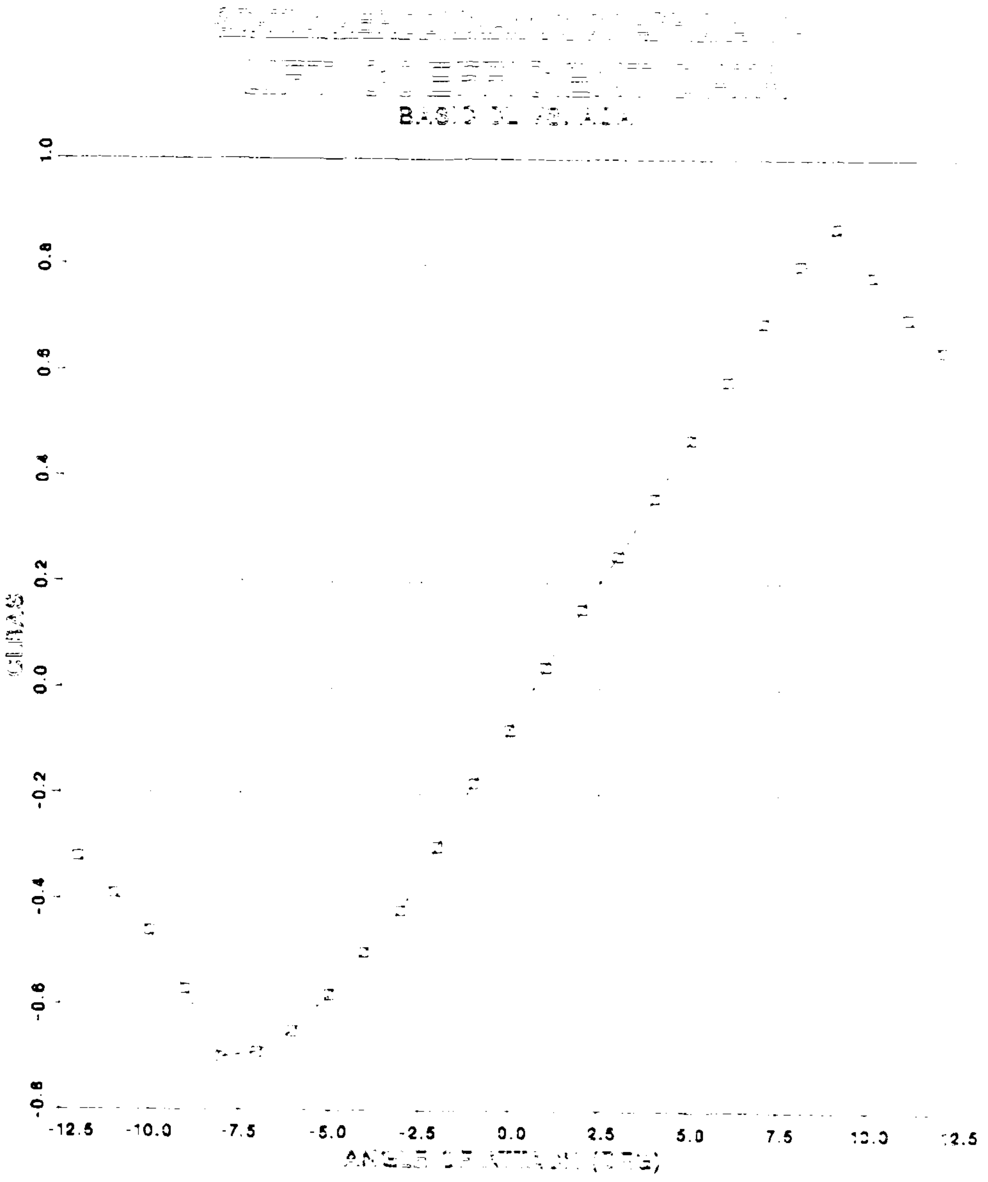


Figure A.2 Data Array LFT1.

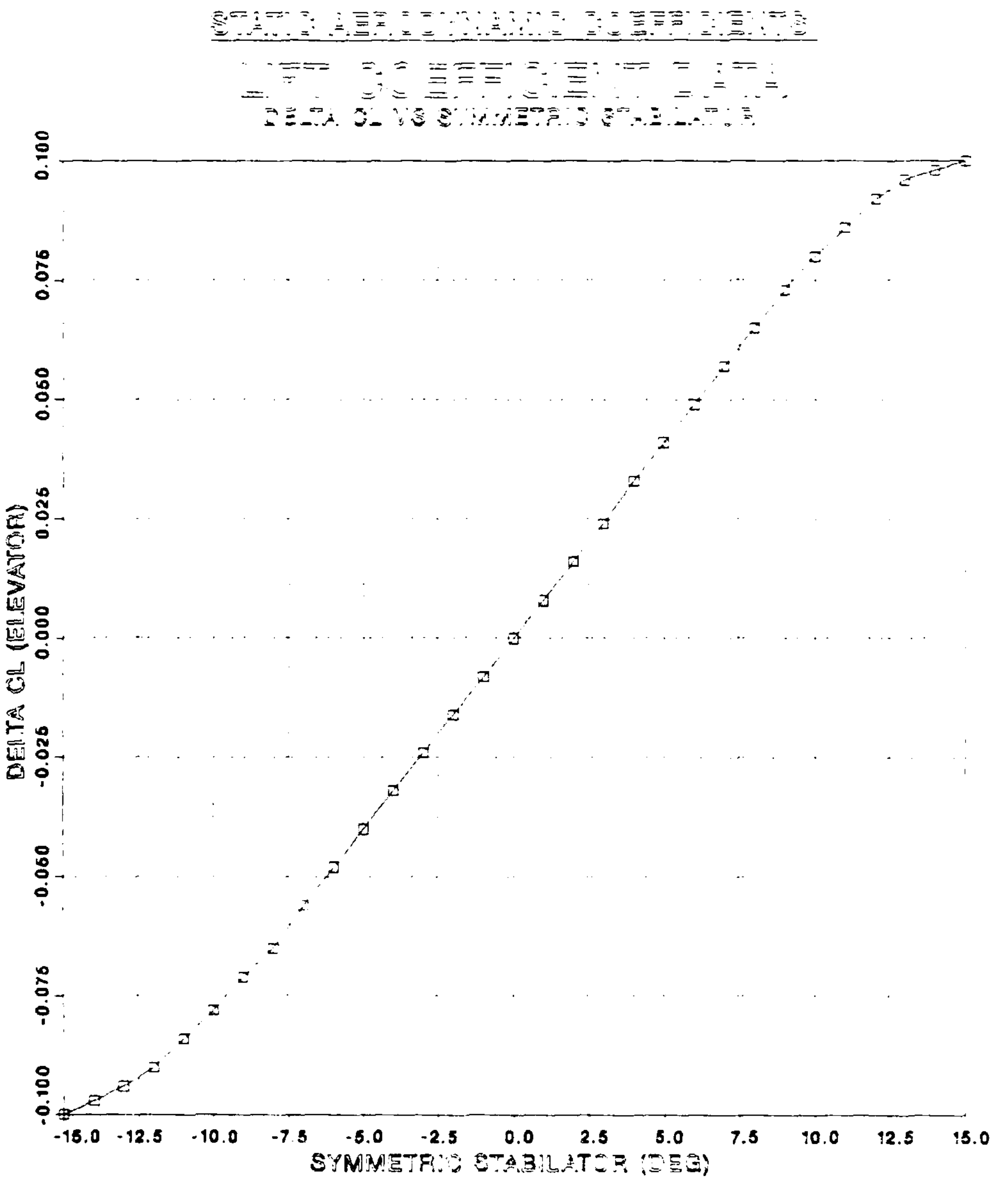


Figure A.3 Data Array LFT2.

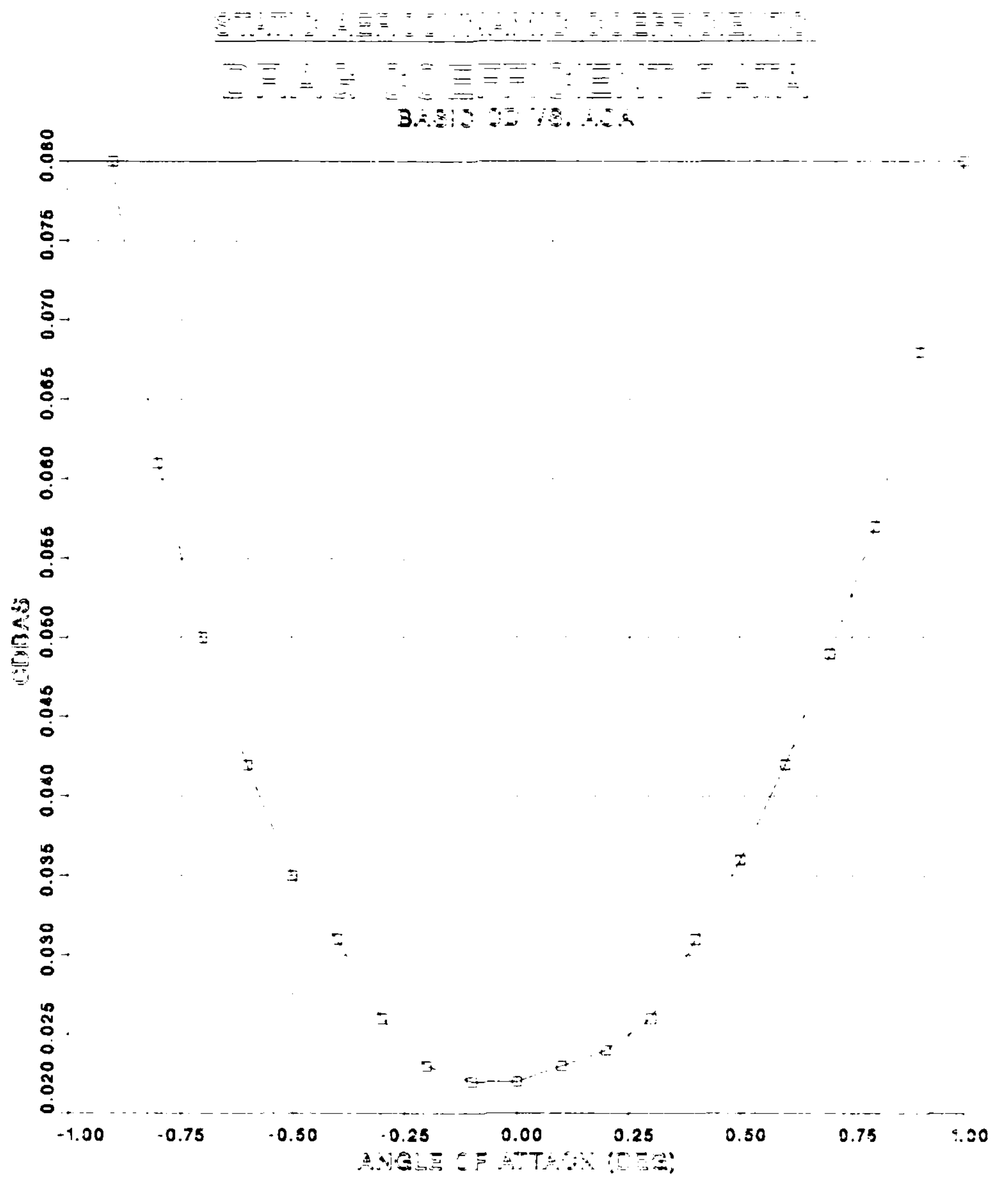


Figure A.4 Data Array DRG1.

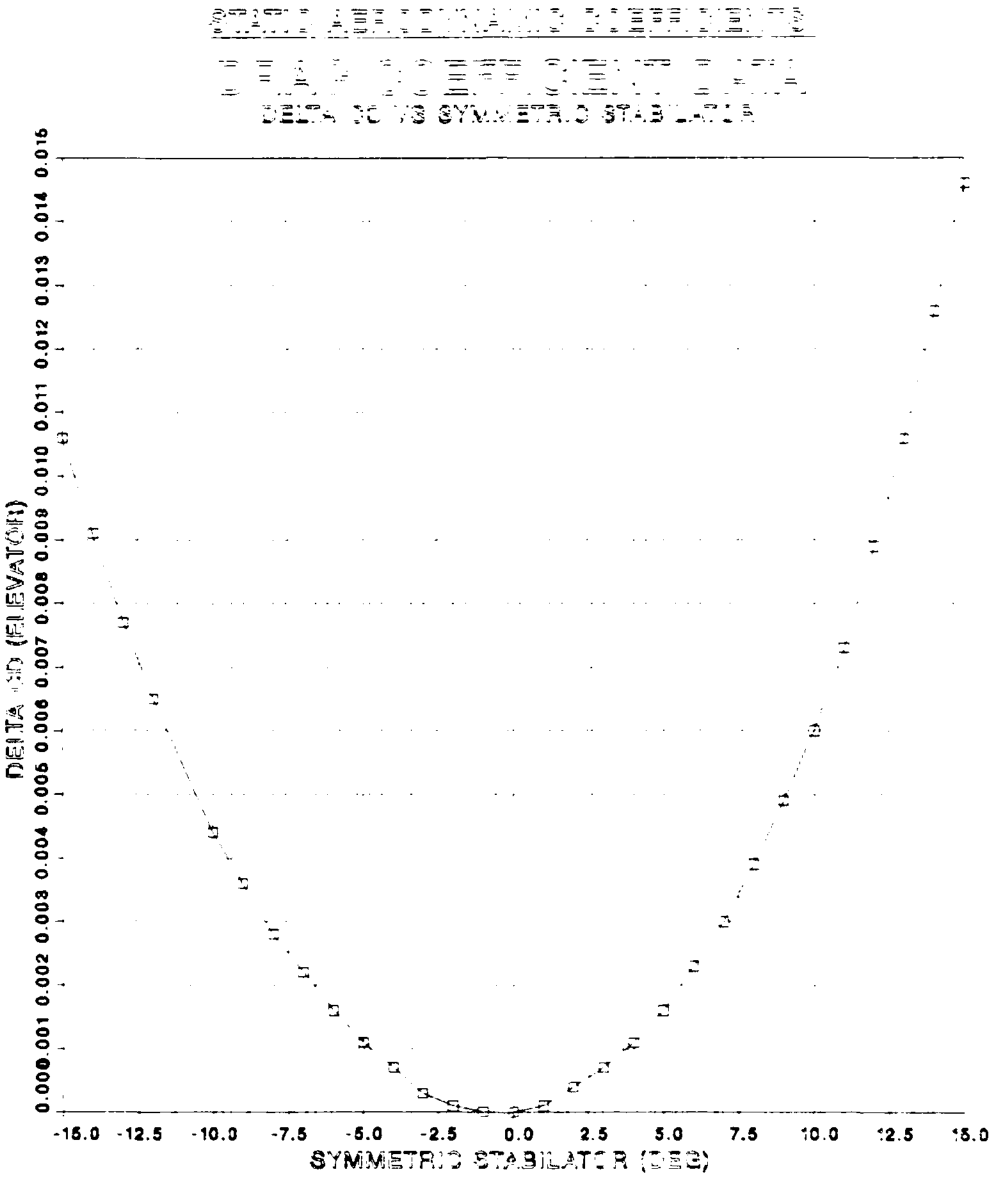


Figure A.5 Data Array DRG2.

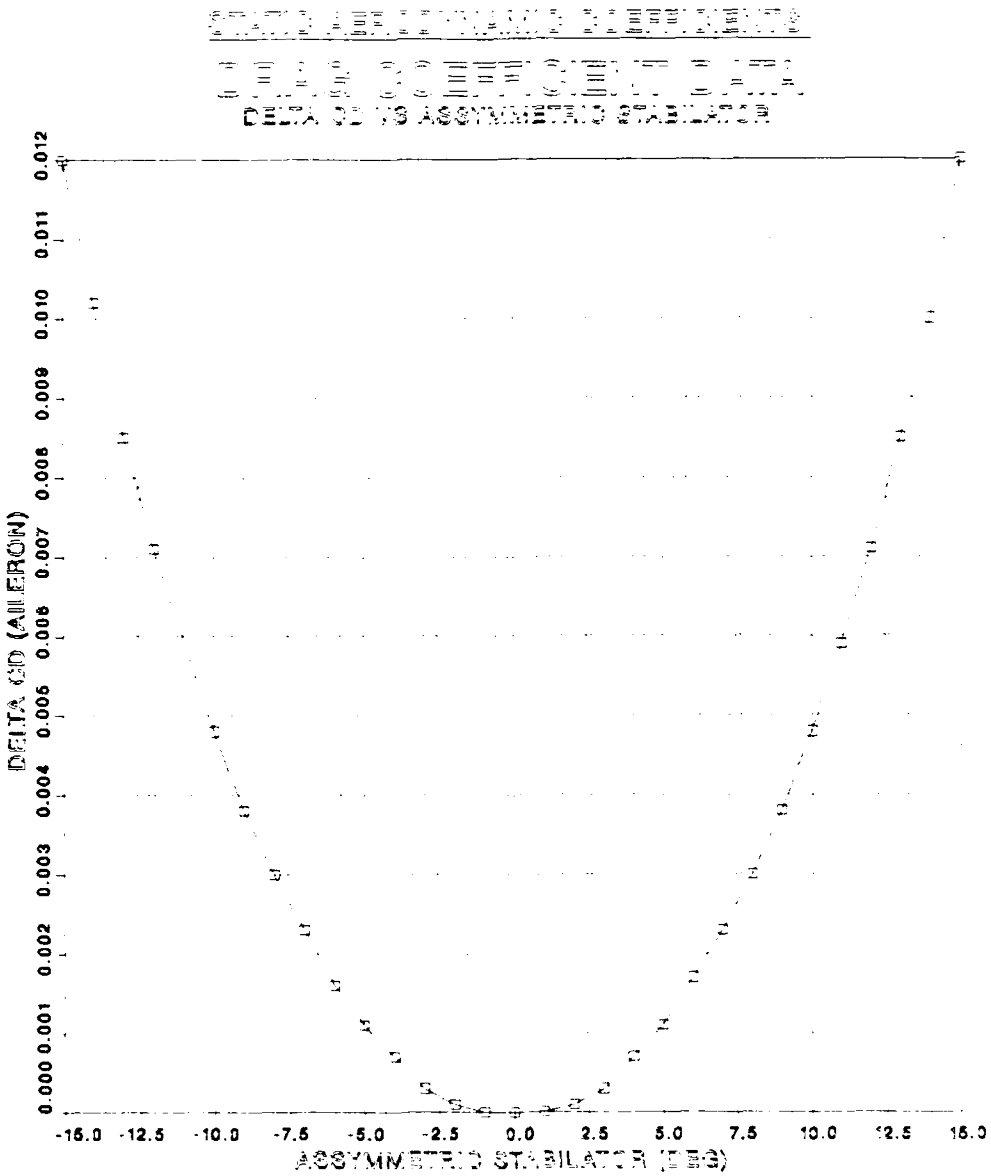


Figure A.6 Data Array DRG3.

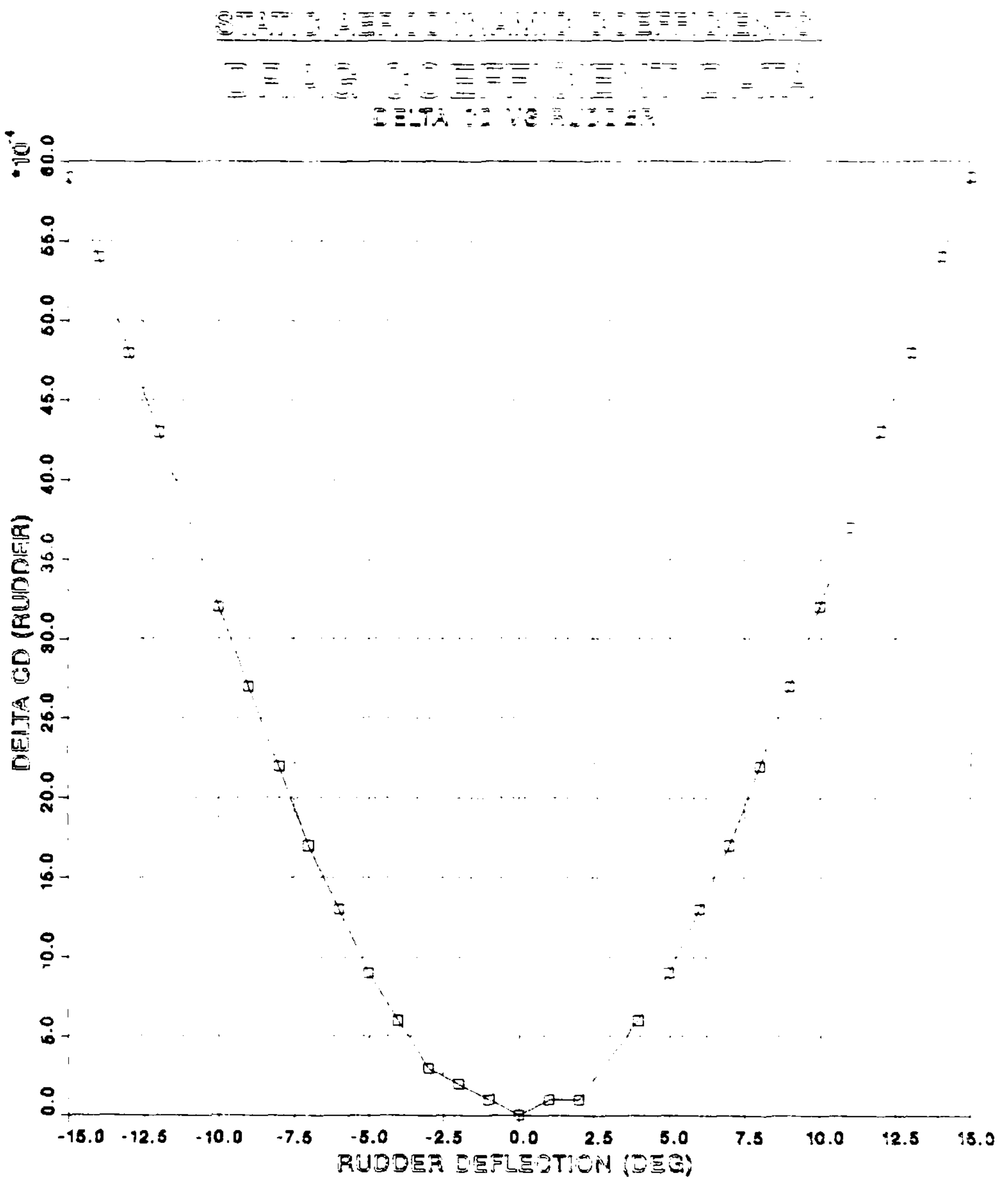


Figure A.7 Dara Array DPG4.

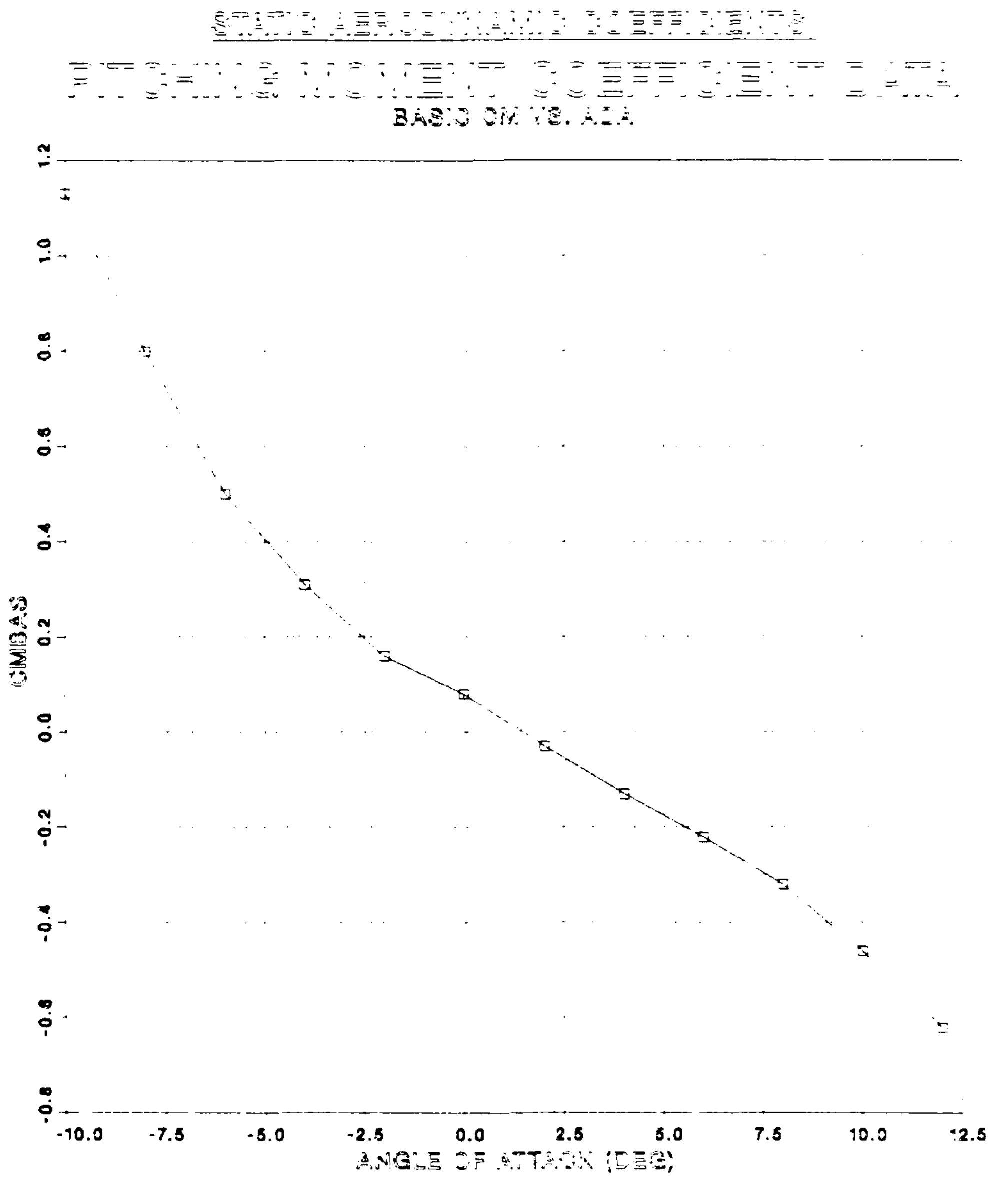


Figure A.8 Data Array PTCH1.

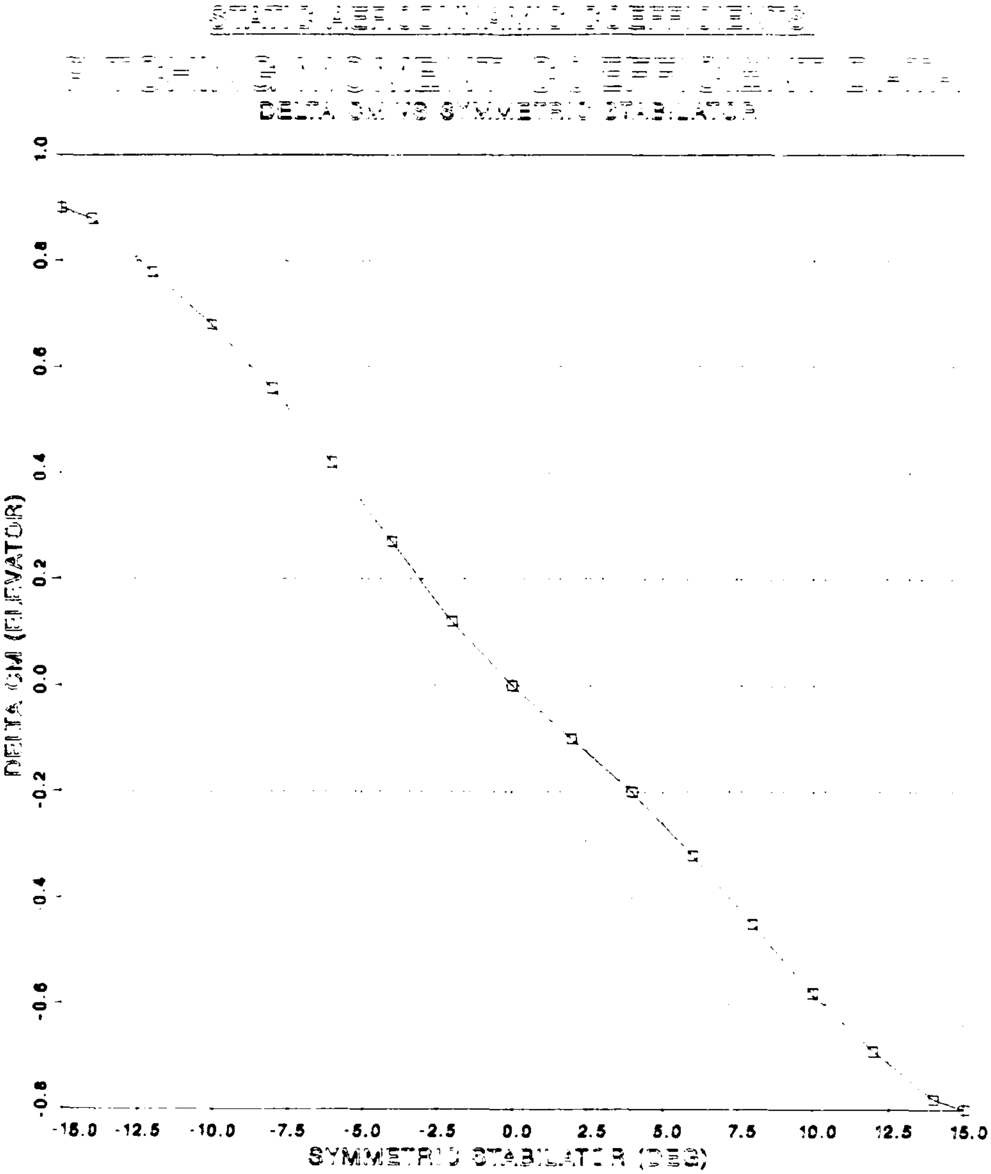


Figure A.9 Data Array PTCH2.

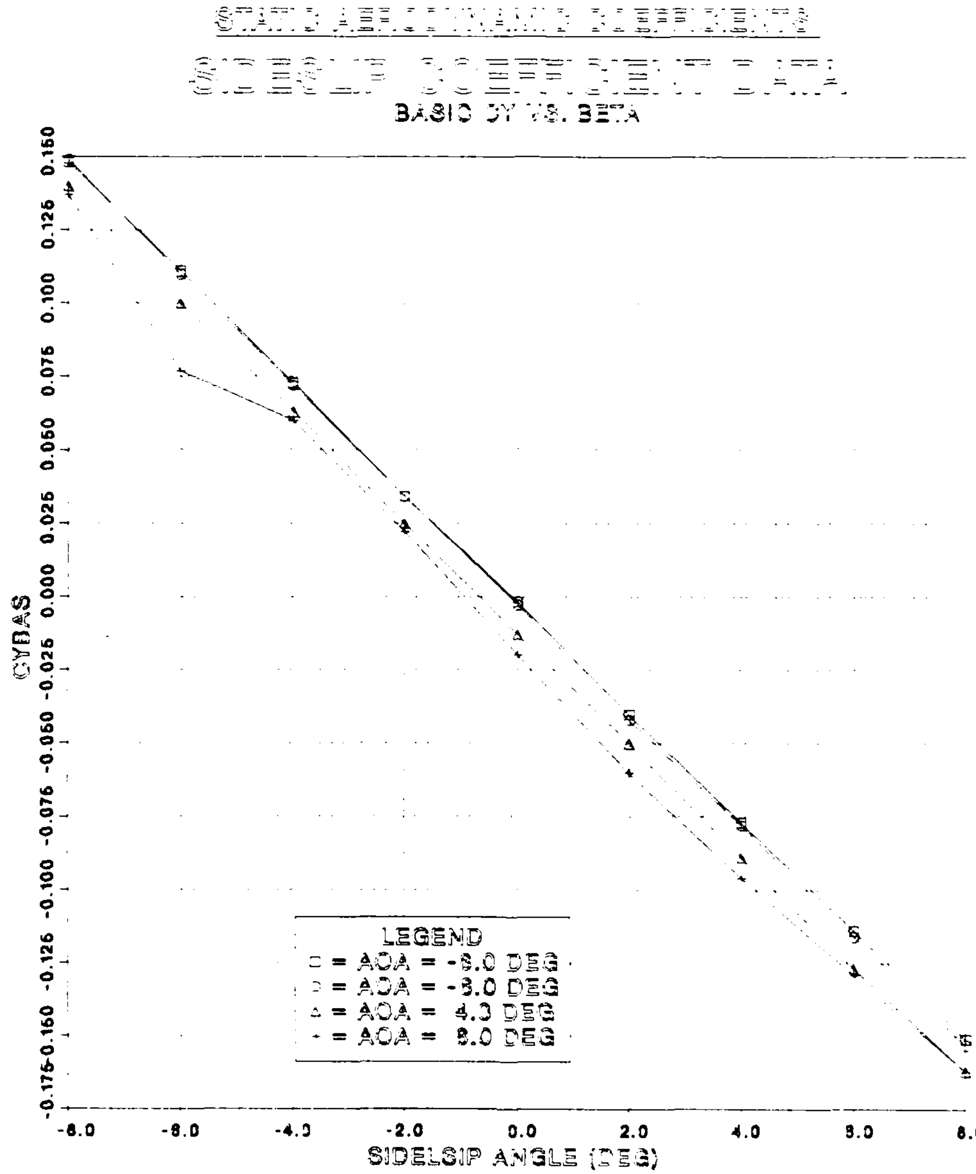


Figure A.10 Data Array SID1.

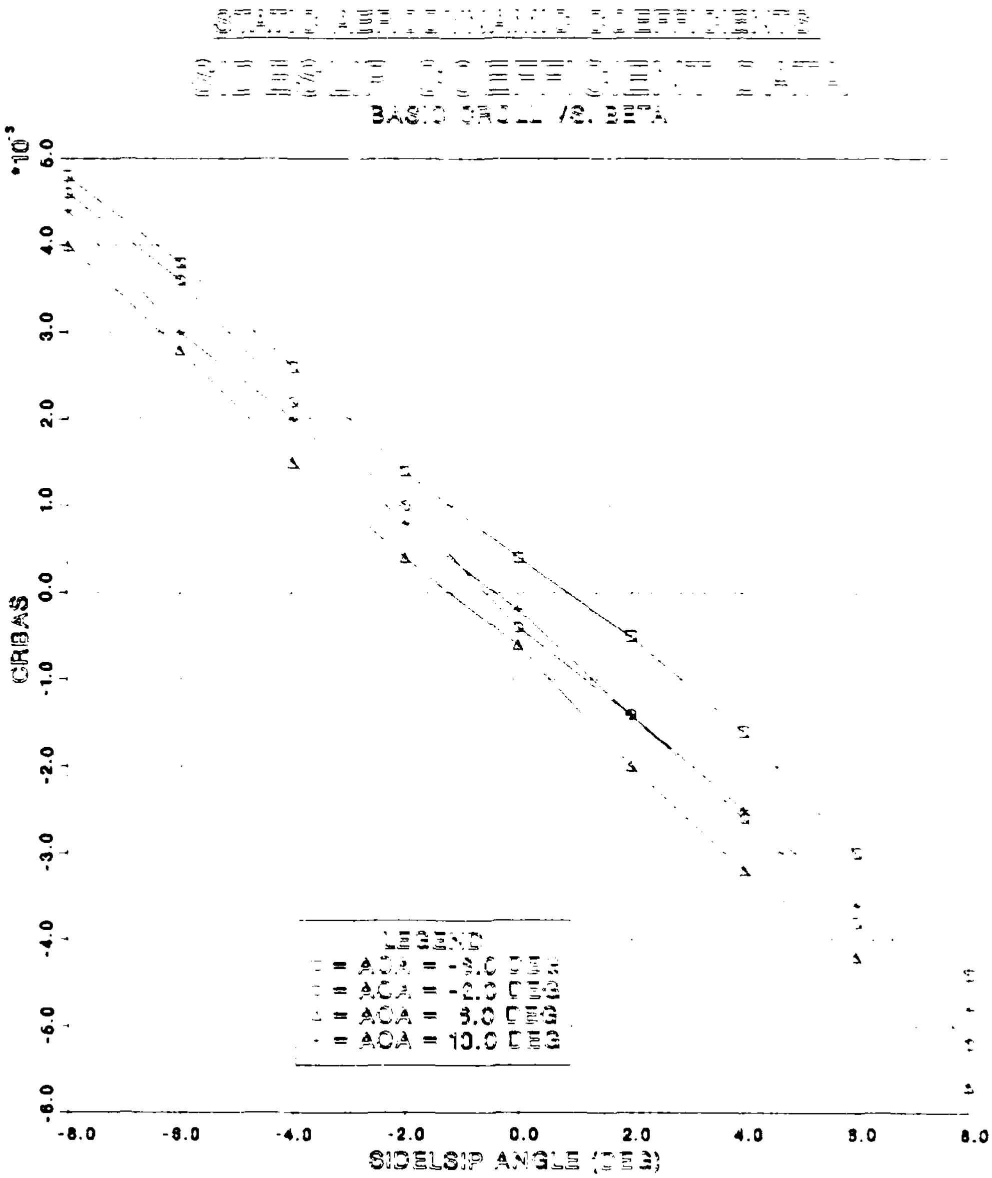


Figure A.11   Data Array SID2.

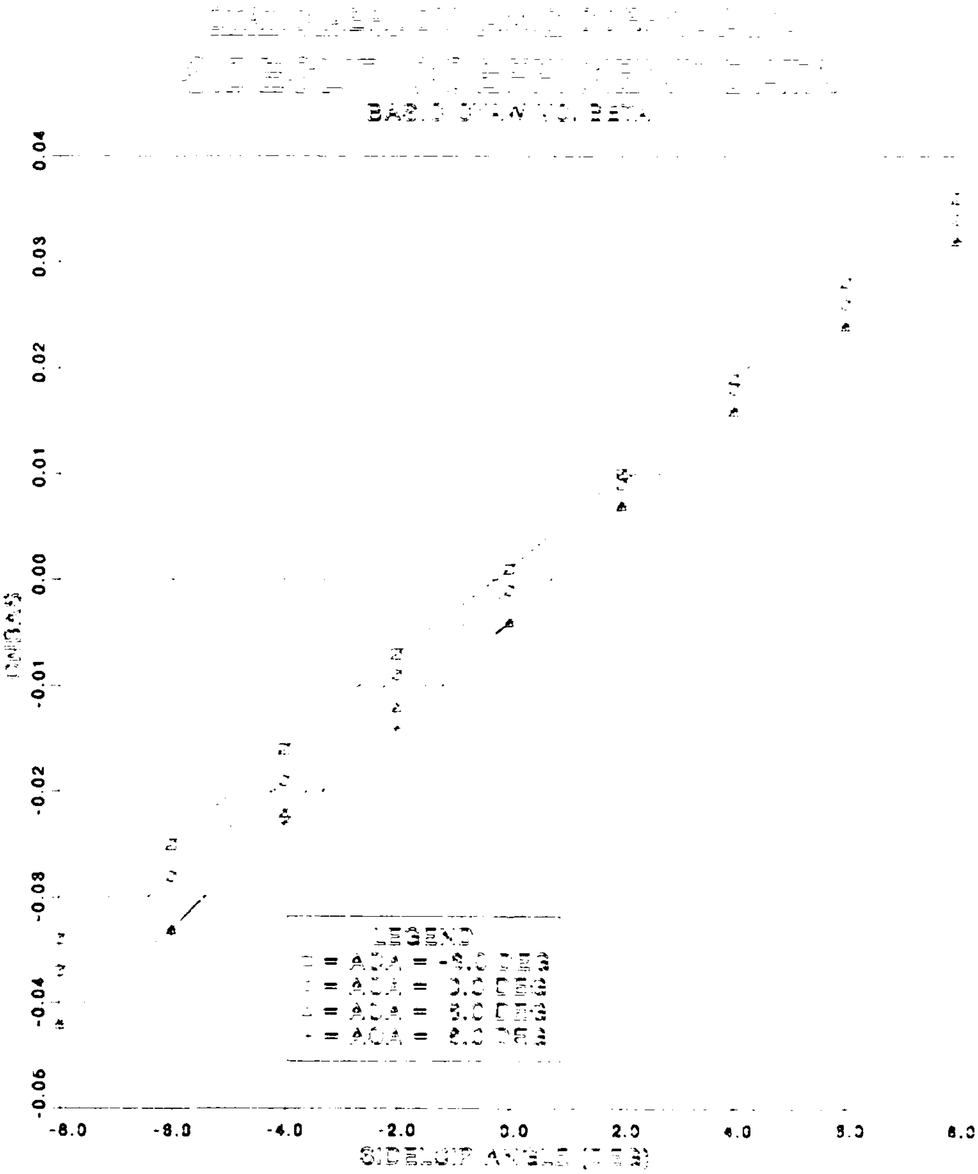


Figure A.12   Data Array SID3.

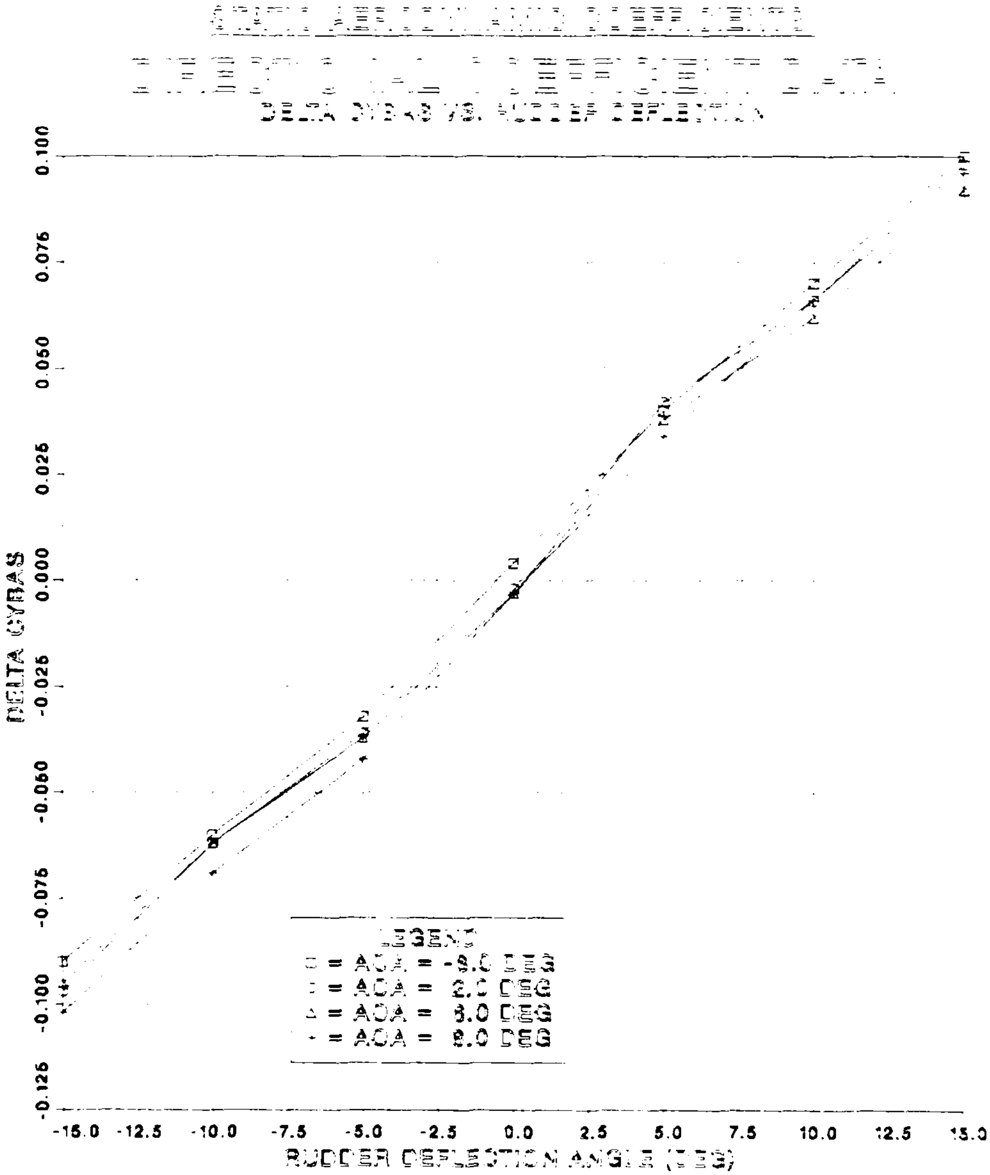


Figure A.13 Data Array DREC1.

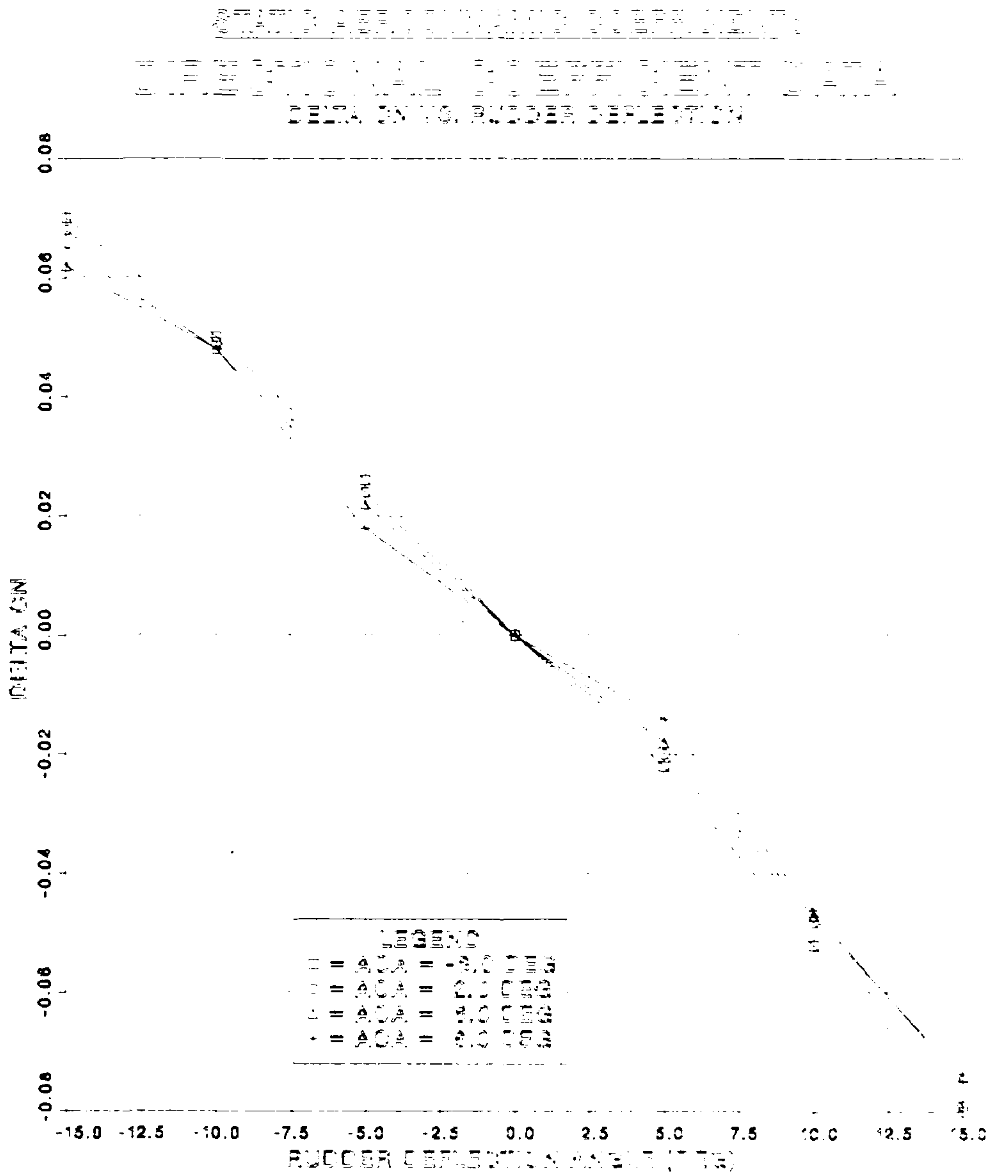


Figure A.14 Data Array DREC2.

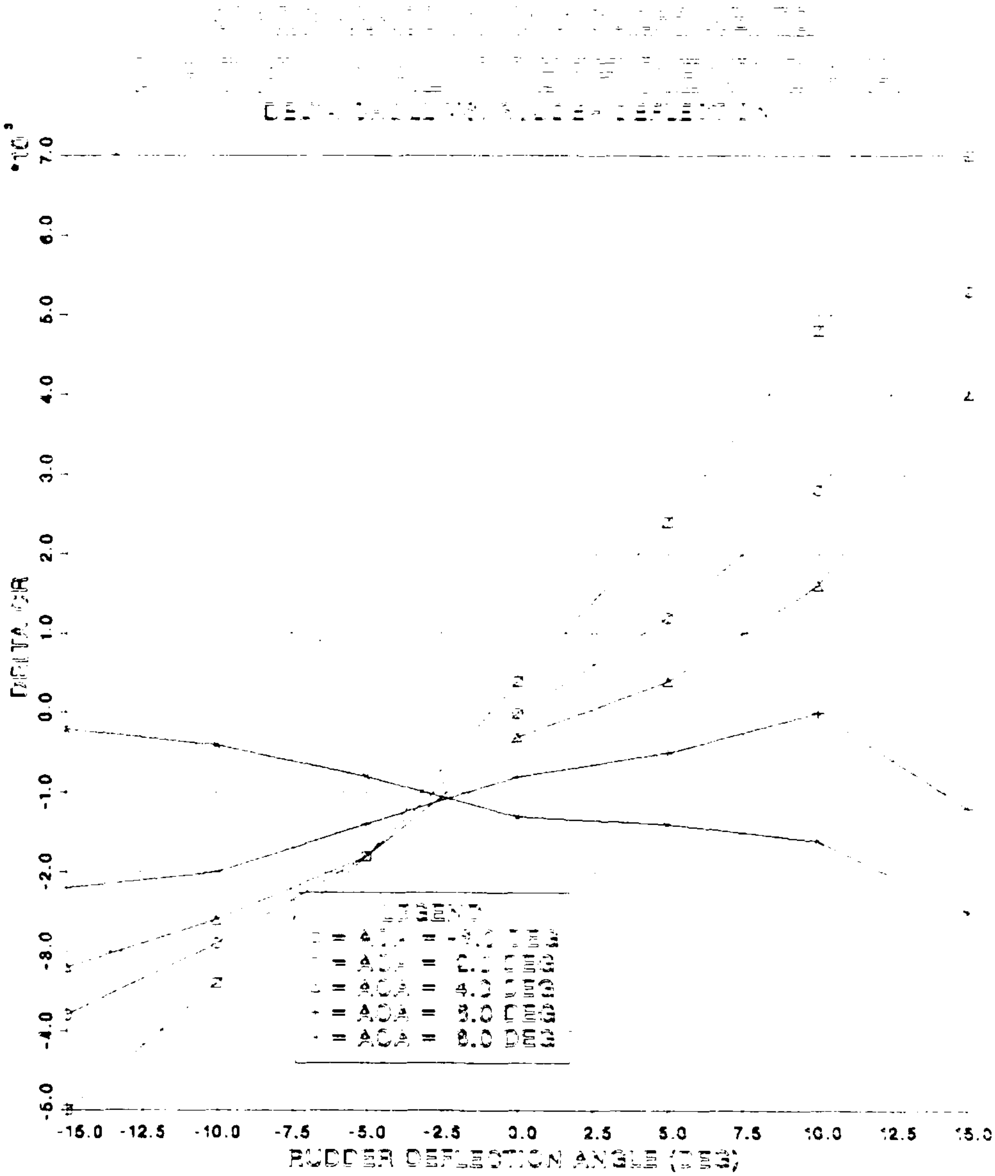


Figure A.15 Data Array DREC3.

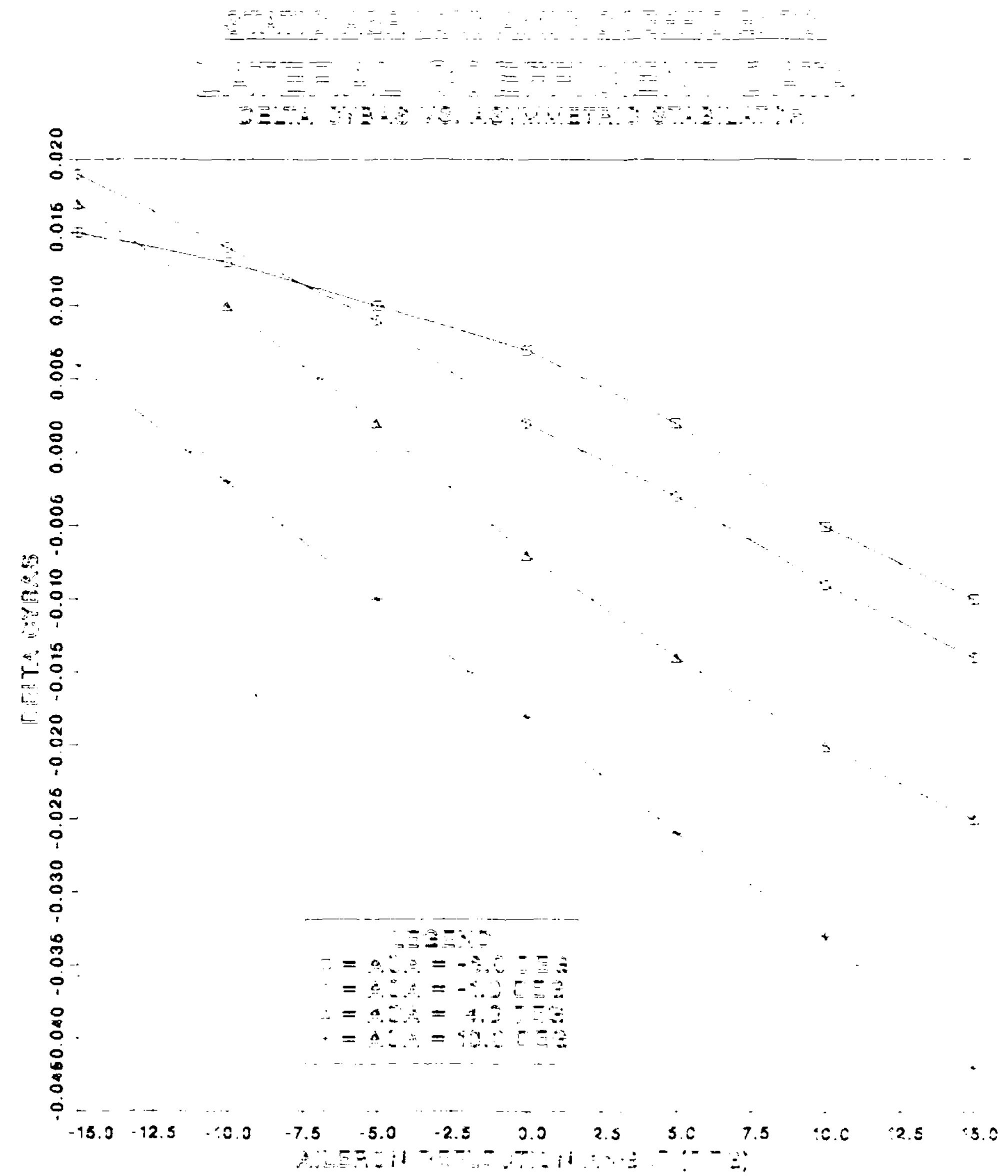


Figure A.16   Data Array LTEL1.

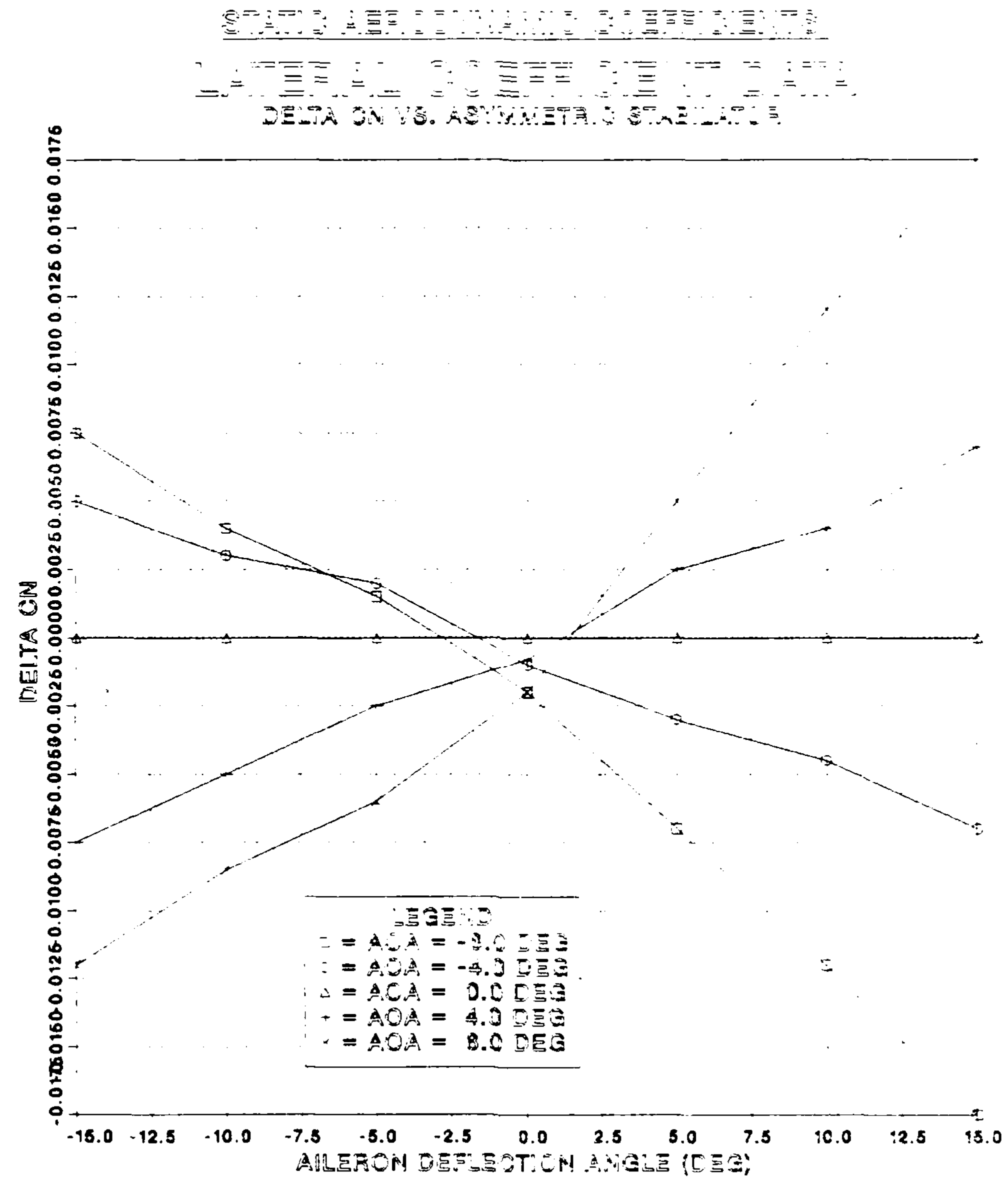


Figure A.17 Data Array LTFL2.

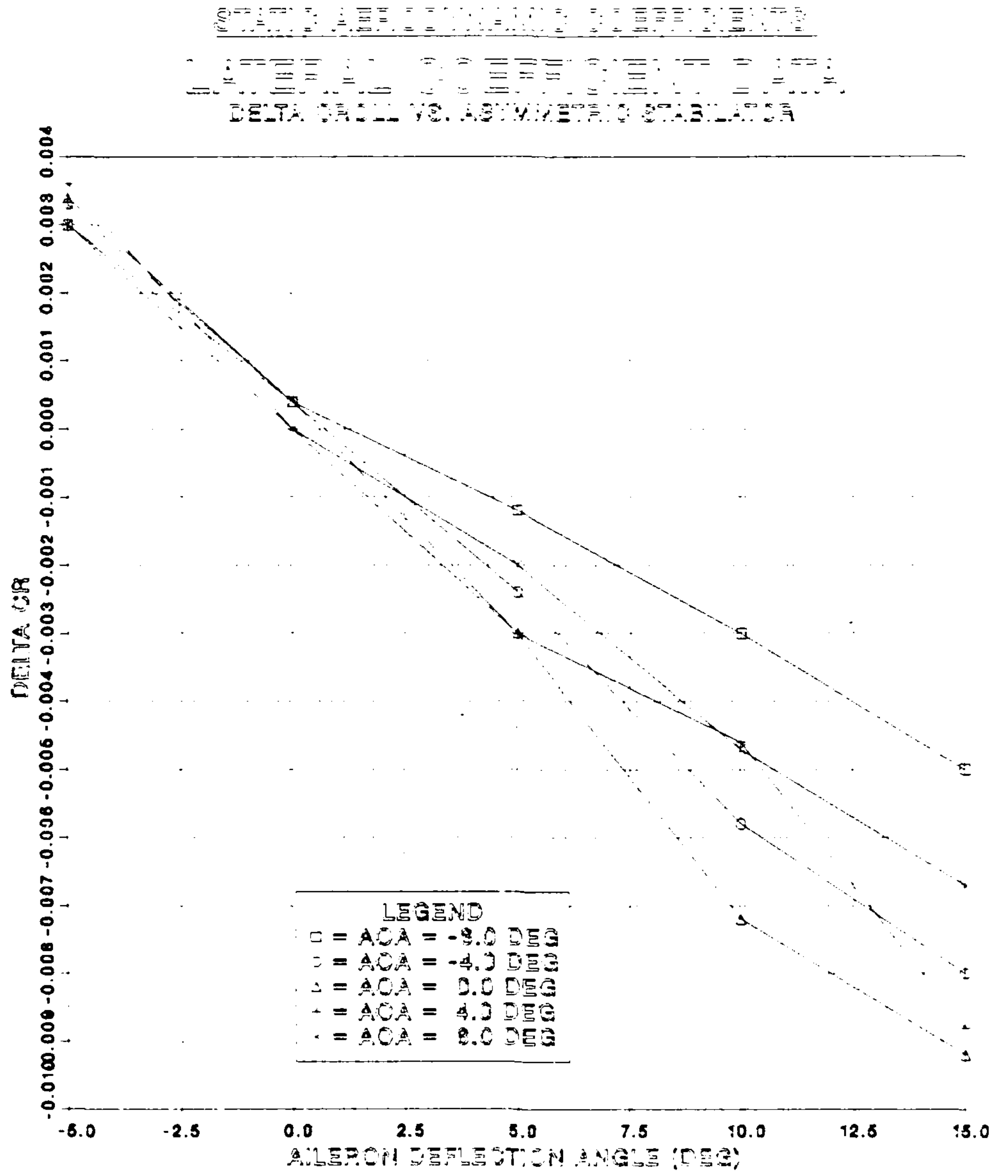


Figure A.18 Data Array LTRL3.

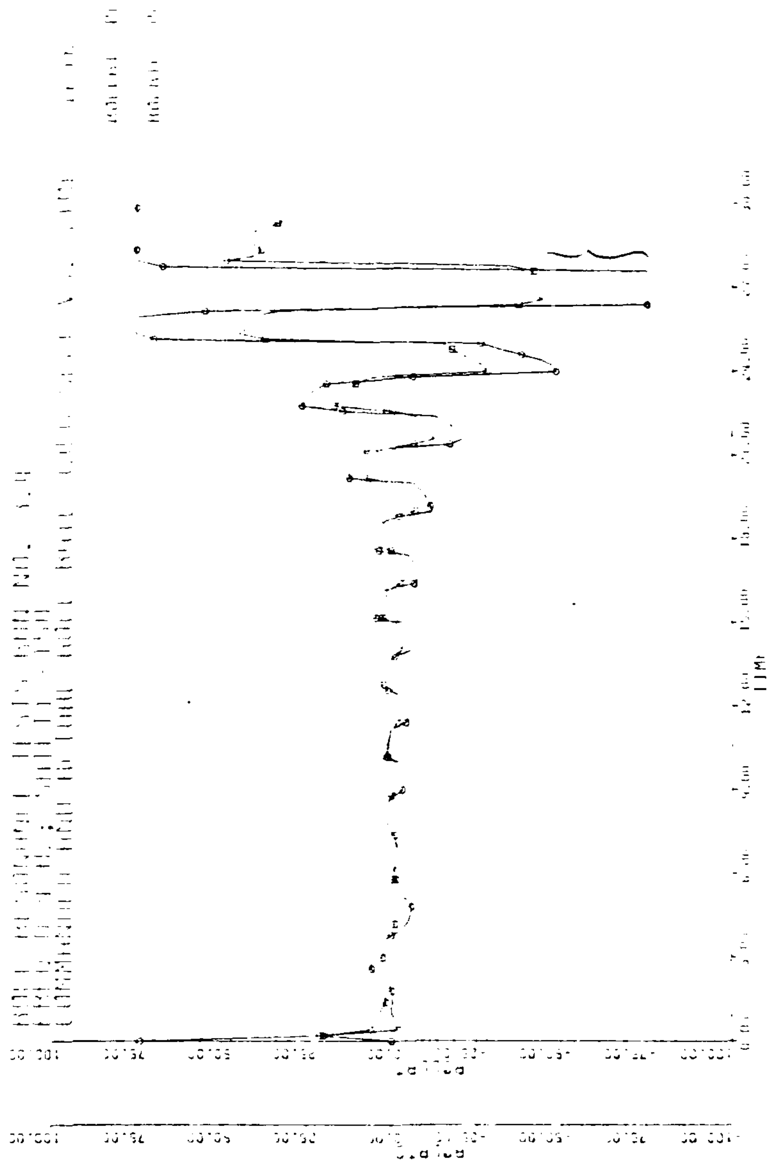


Figure A.19 CSMP Data (Roll Rate) - KROLLR = 0.1.

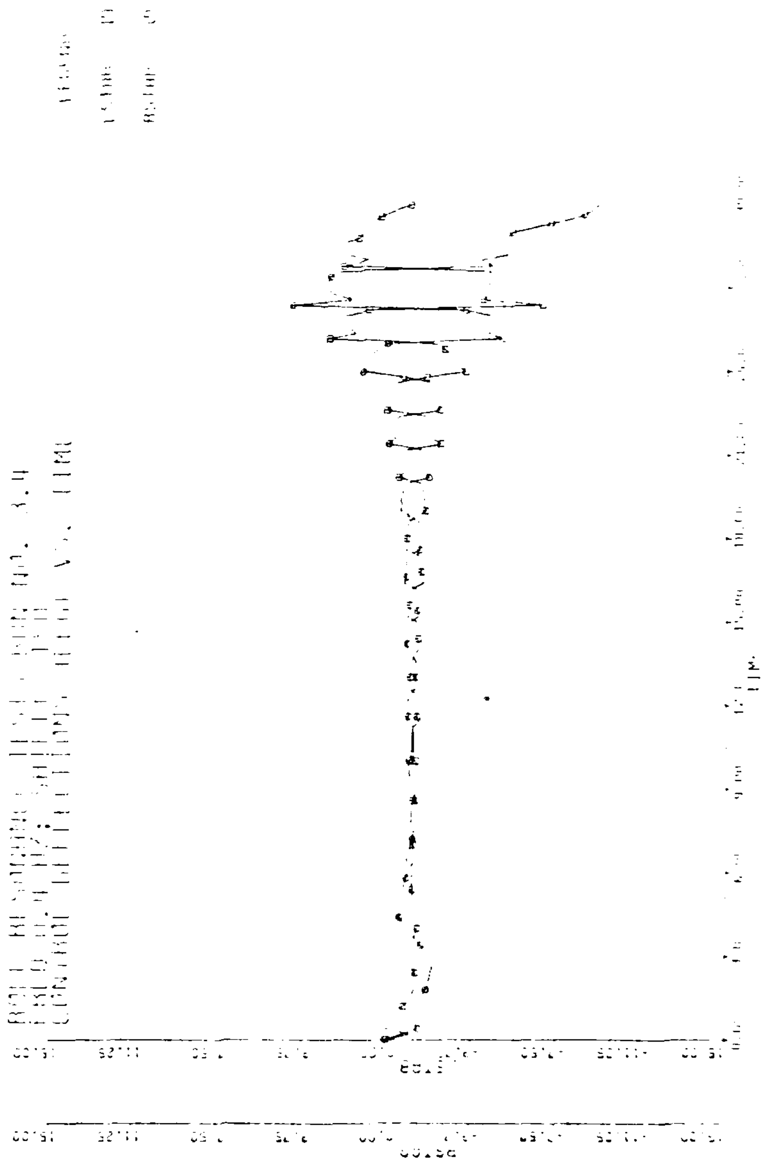


Figure A.20   CSMP Data (Controls) - KROLLF = 0.1.

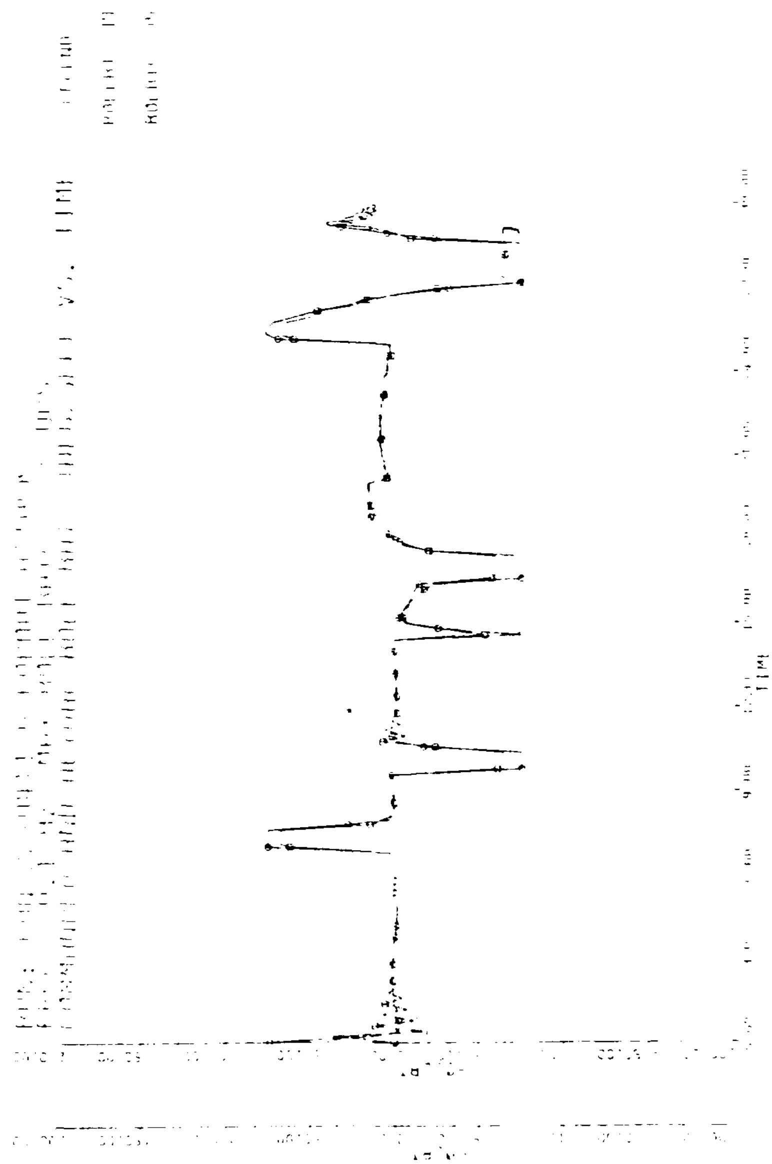


Figure A.21 CSMP Data (Roll Rate) - KROLLP = 0.5.

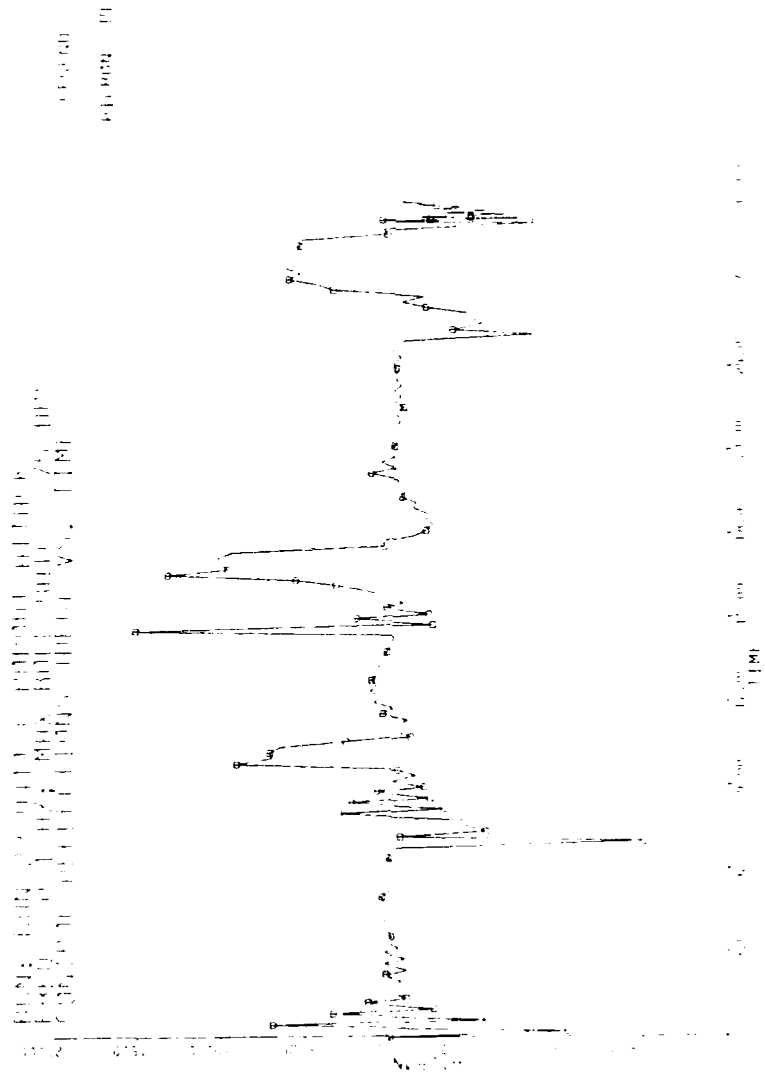


Figure A.22   CSMP Data (Controls) - Krollr = 0.5.

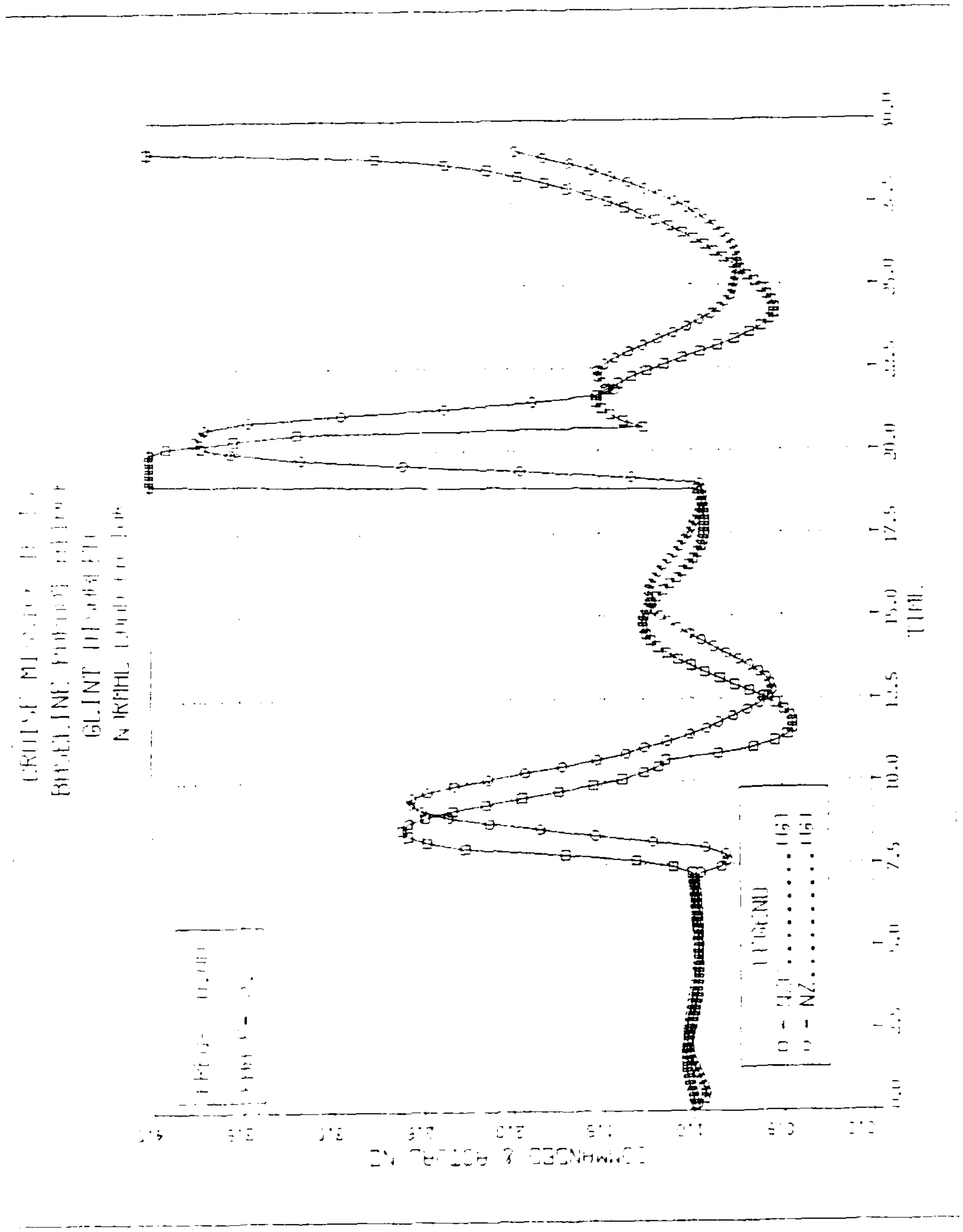


Figure A.23 Baseline - no ECM or GLINT - Load Factor.

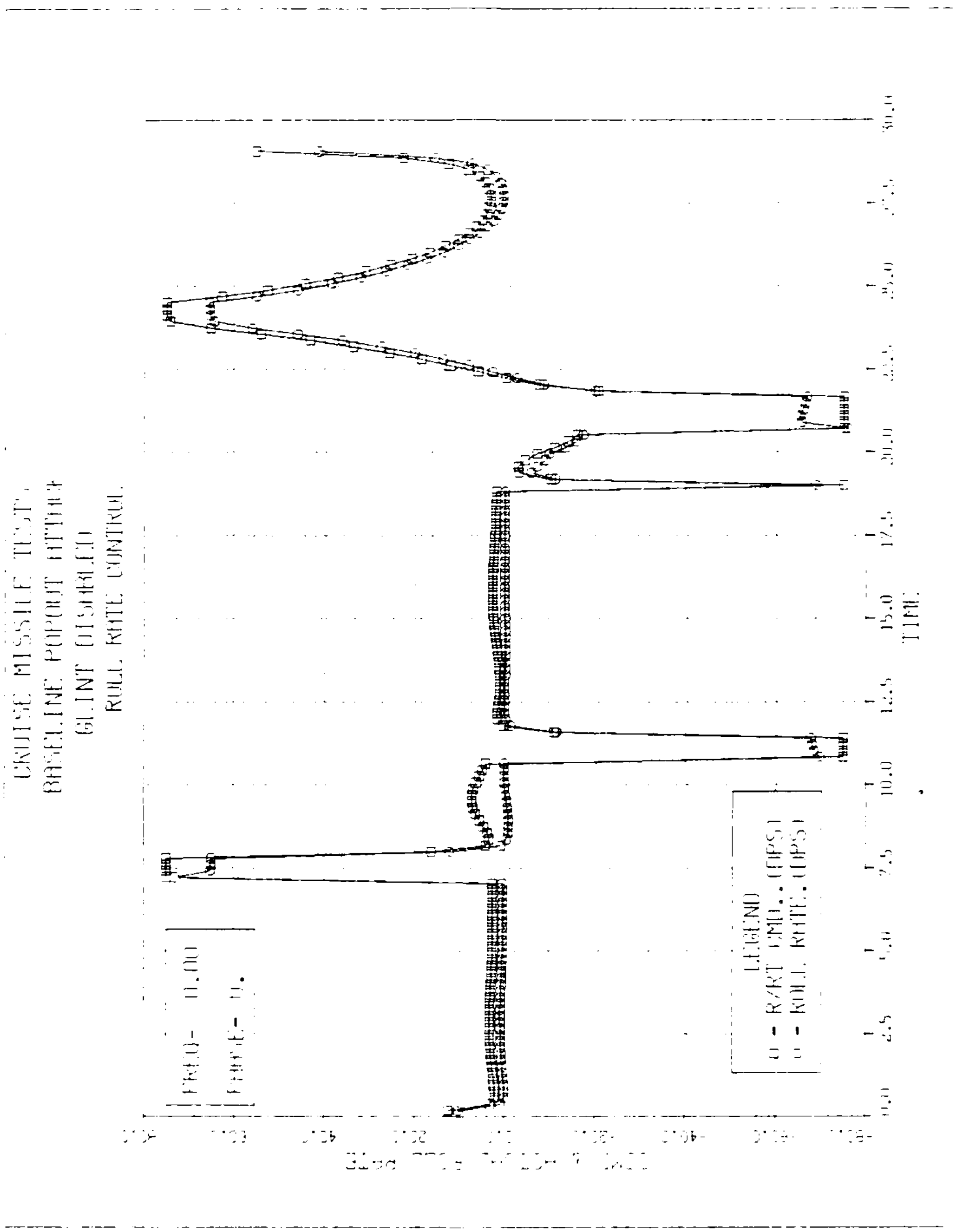


Figure A.24 Baseline - no ECM or GLINT - Roll Rate.

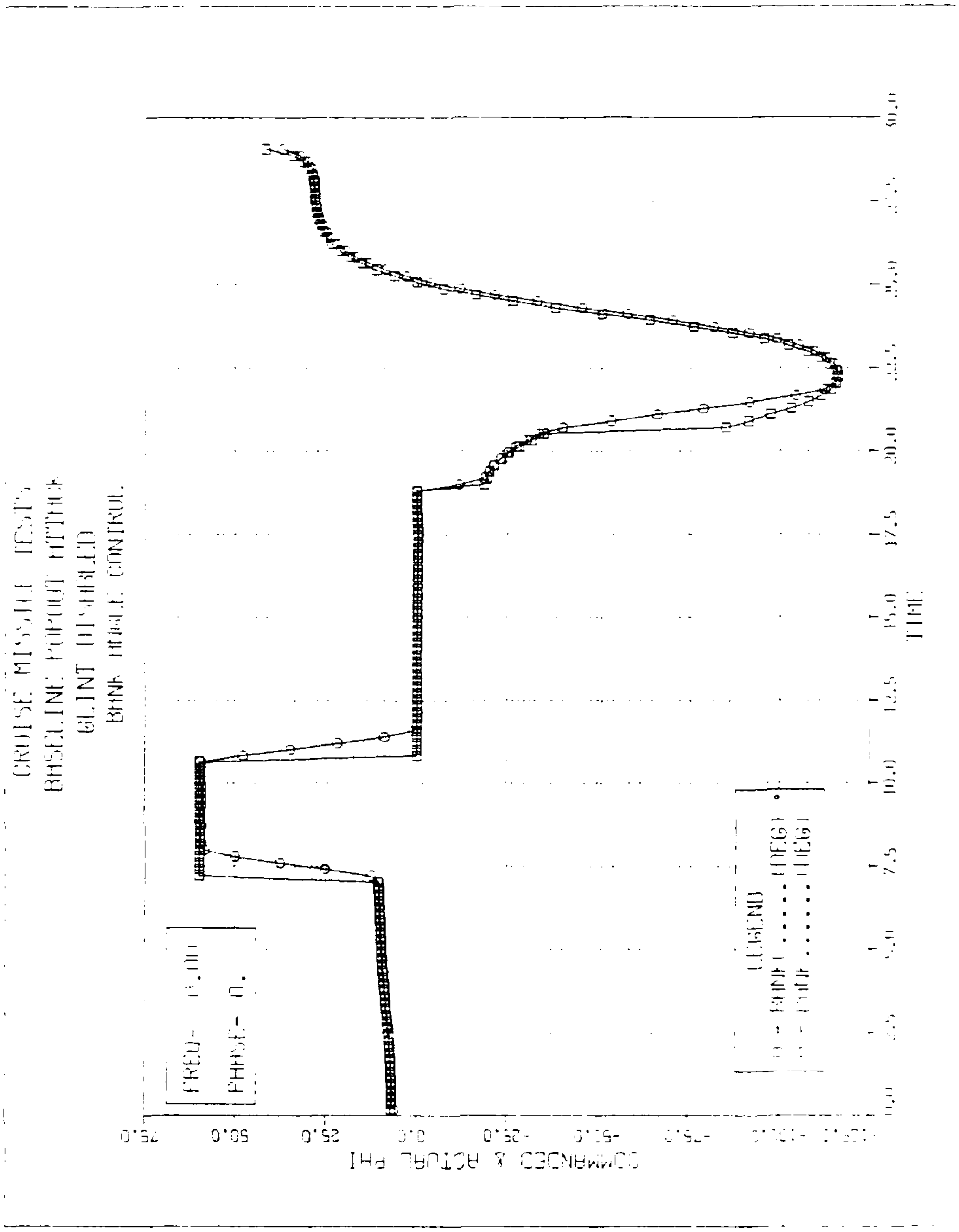


Figure A.25 Baseline - no ECM or GLINT - Bank.

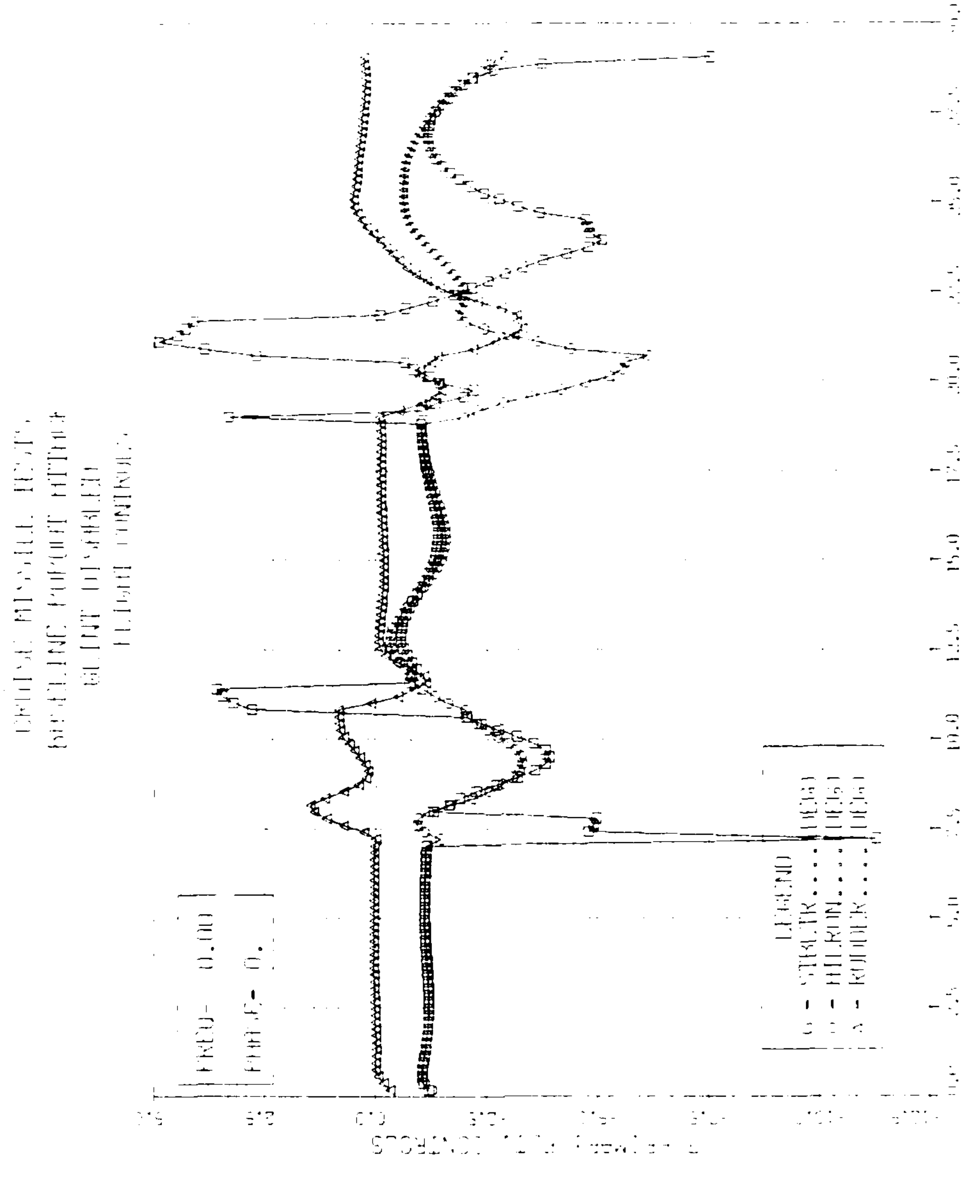


Figure A.26   Baseline - no ECM or GLINT - Controls.

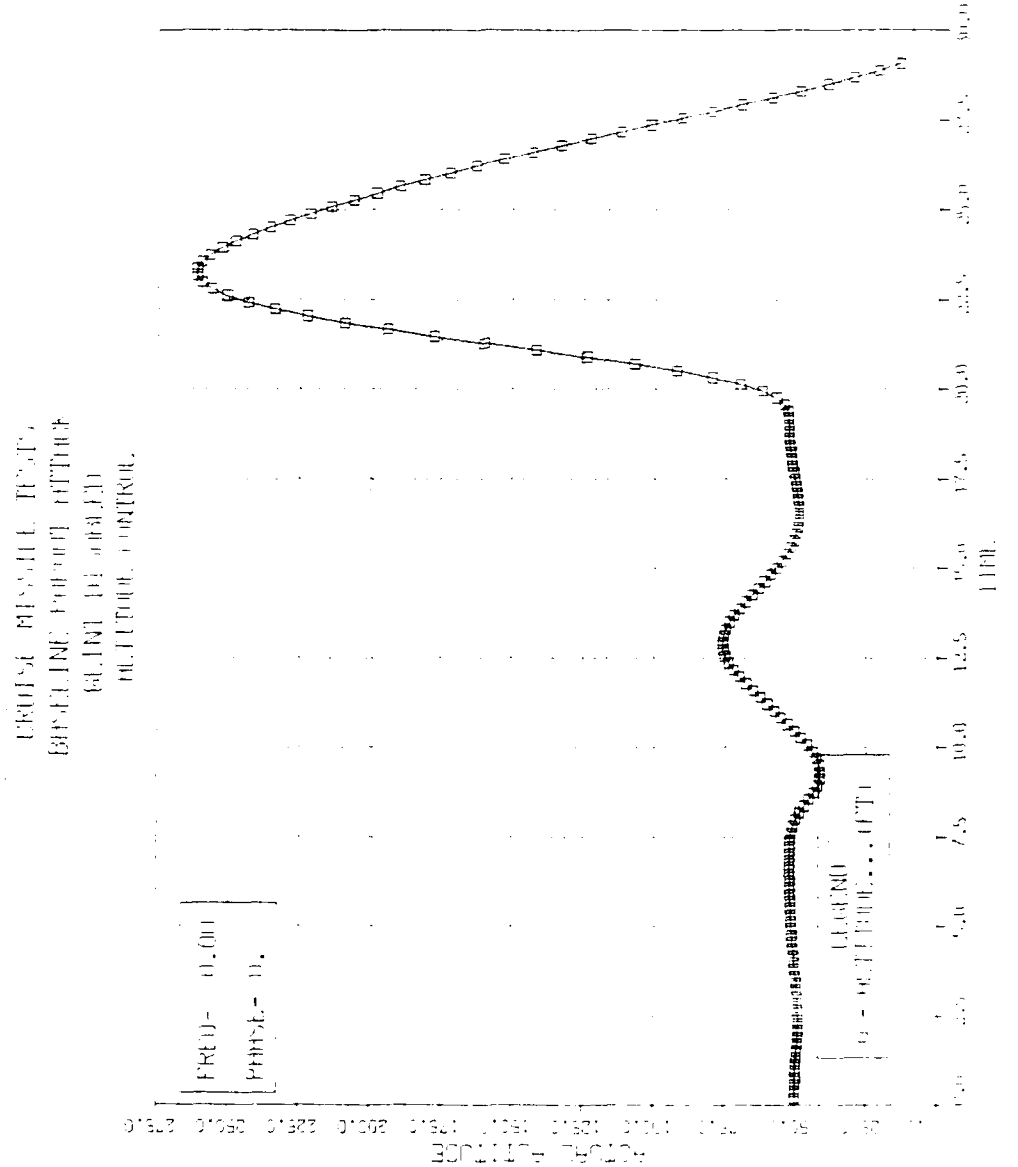


Figure A.27 Baseline - no ECM or GLINT - Altitude.

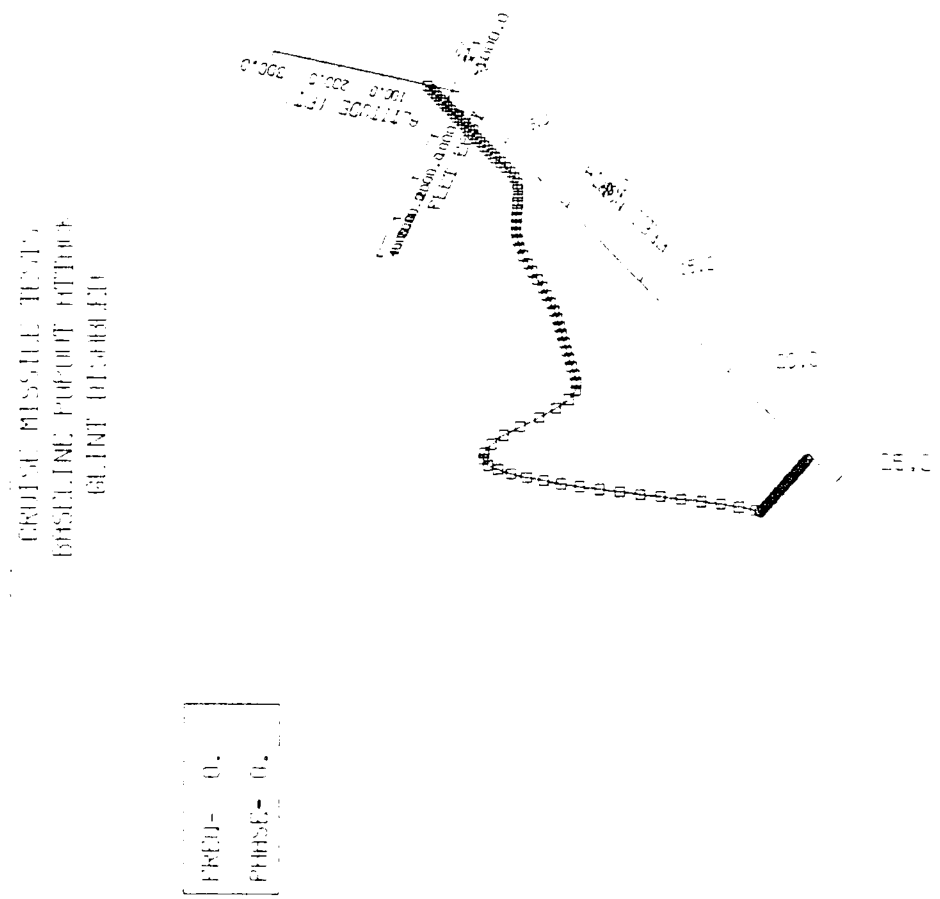


Figure A.28 Baseline - no ECM or GLINT - Geo Plot.

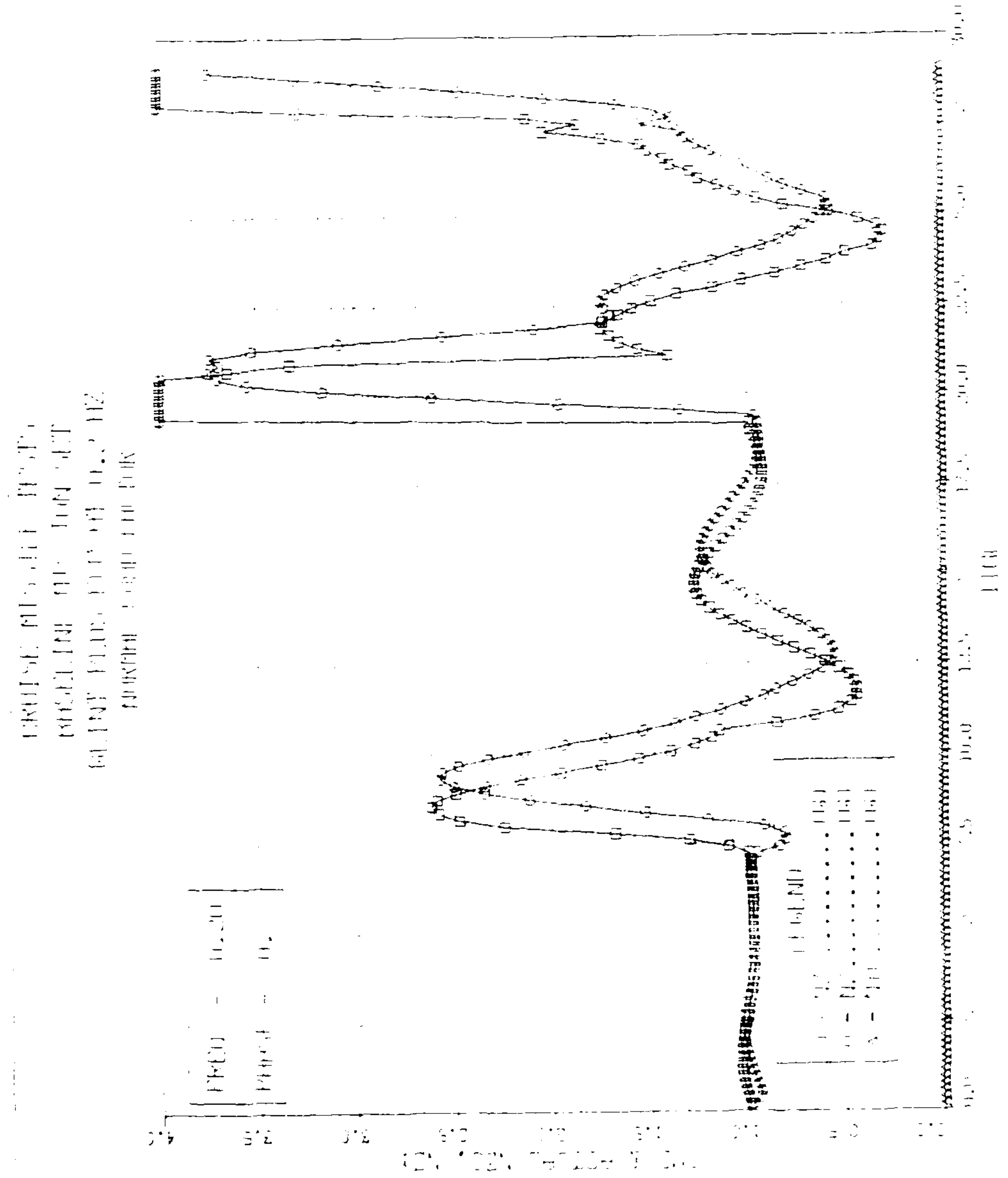


Figure A.29 Baseline with GLINT & ECM - Load Factor.

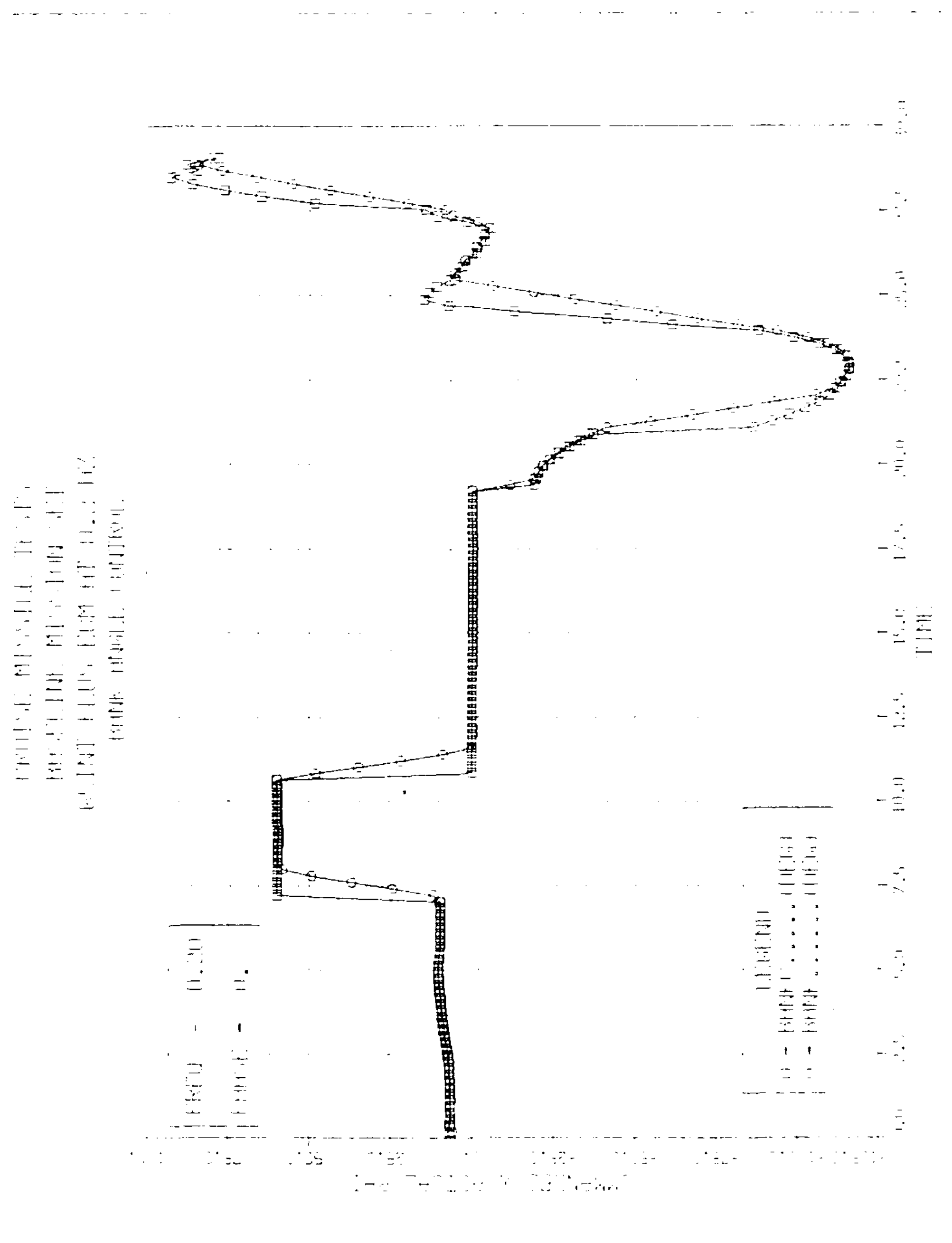


Figure A.30 Baseline with GLINT & ECM - Bank.

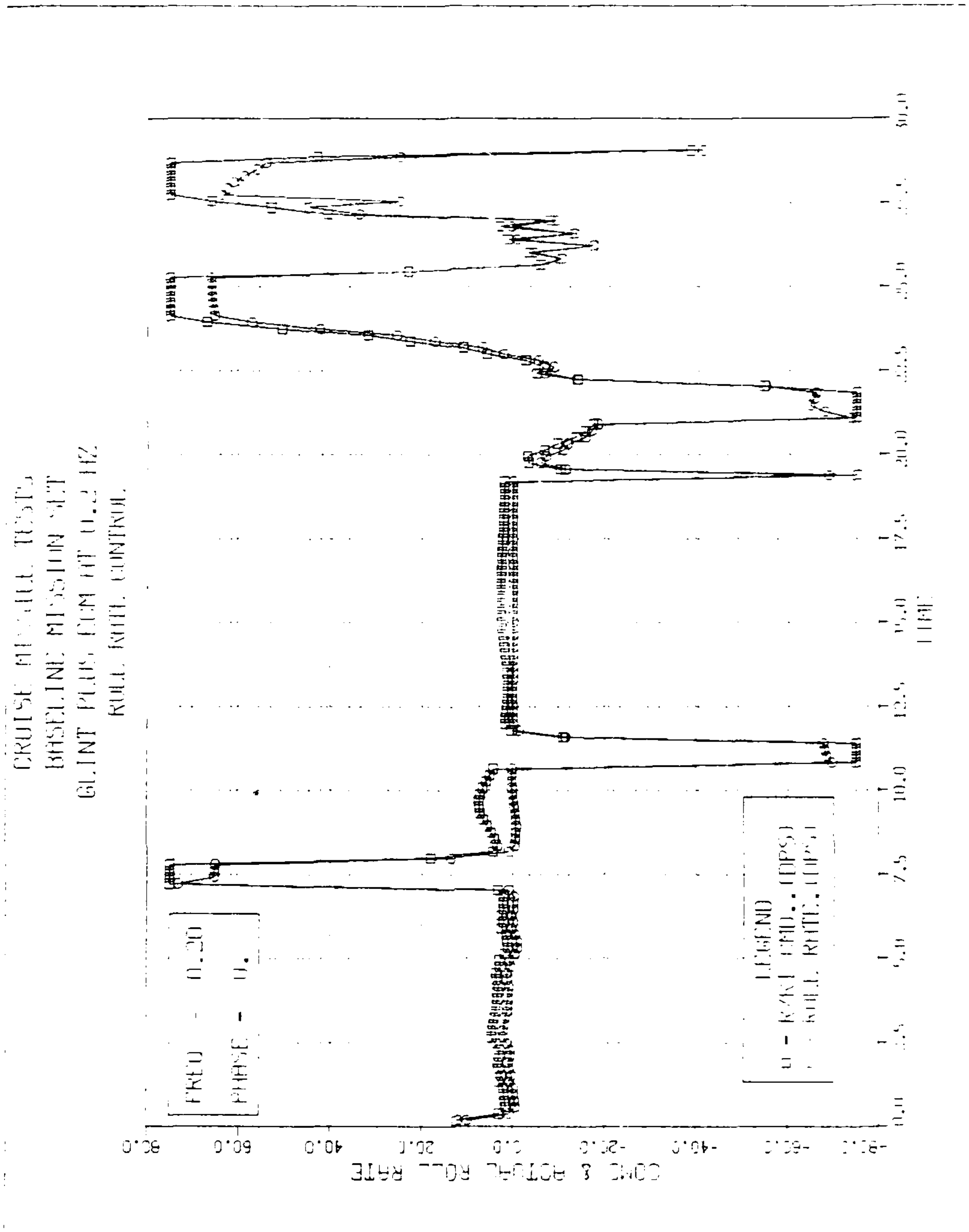


Figure A.31 Baseline with GLINT & ECM - Roll Rate.

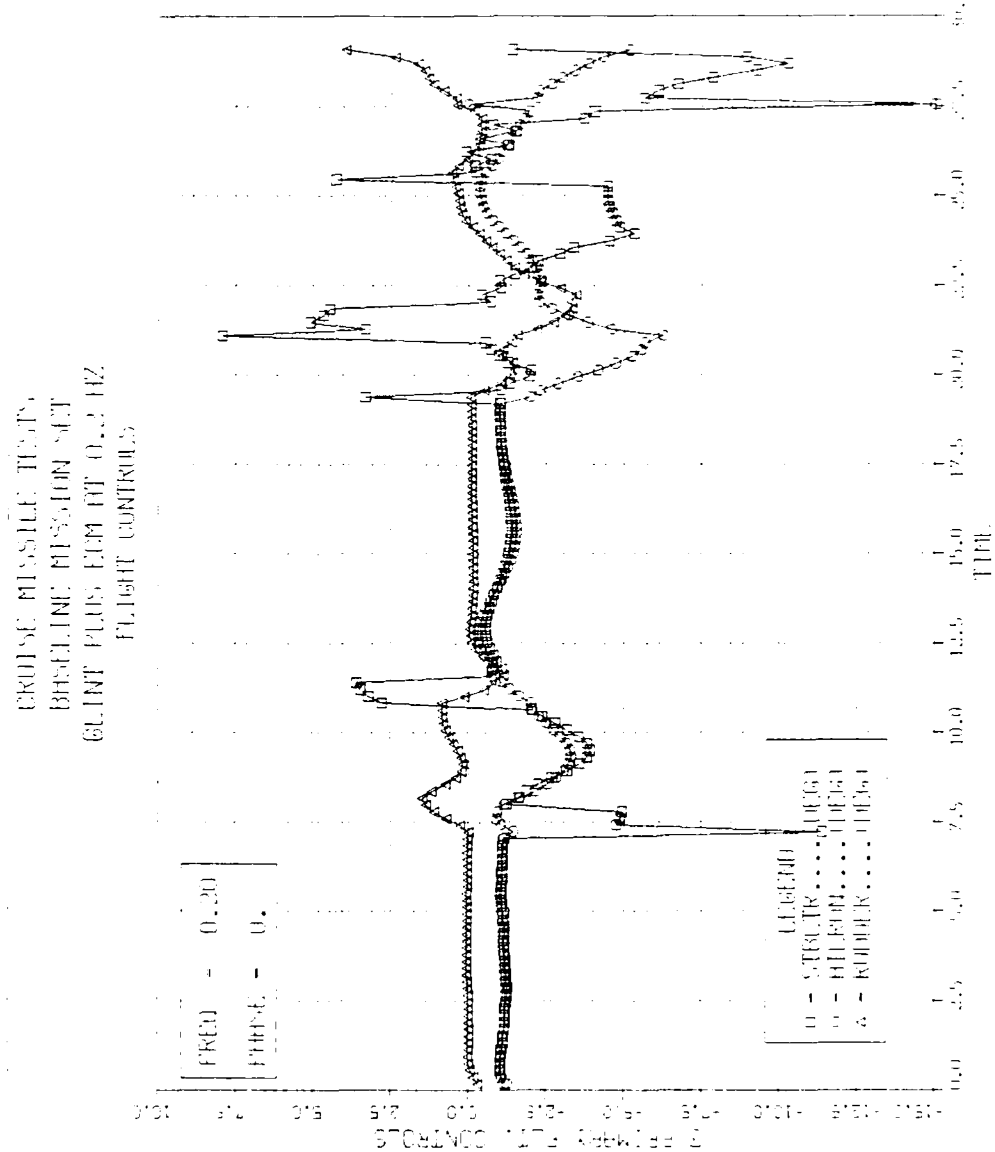


Figure A.32 Baseline with GLINT & ECM - Controls.

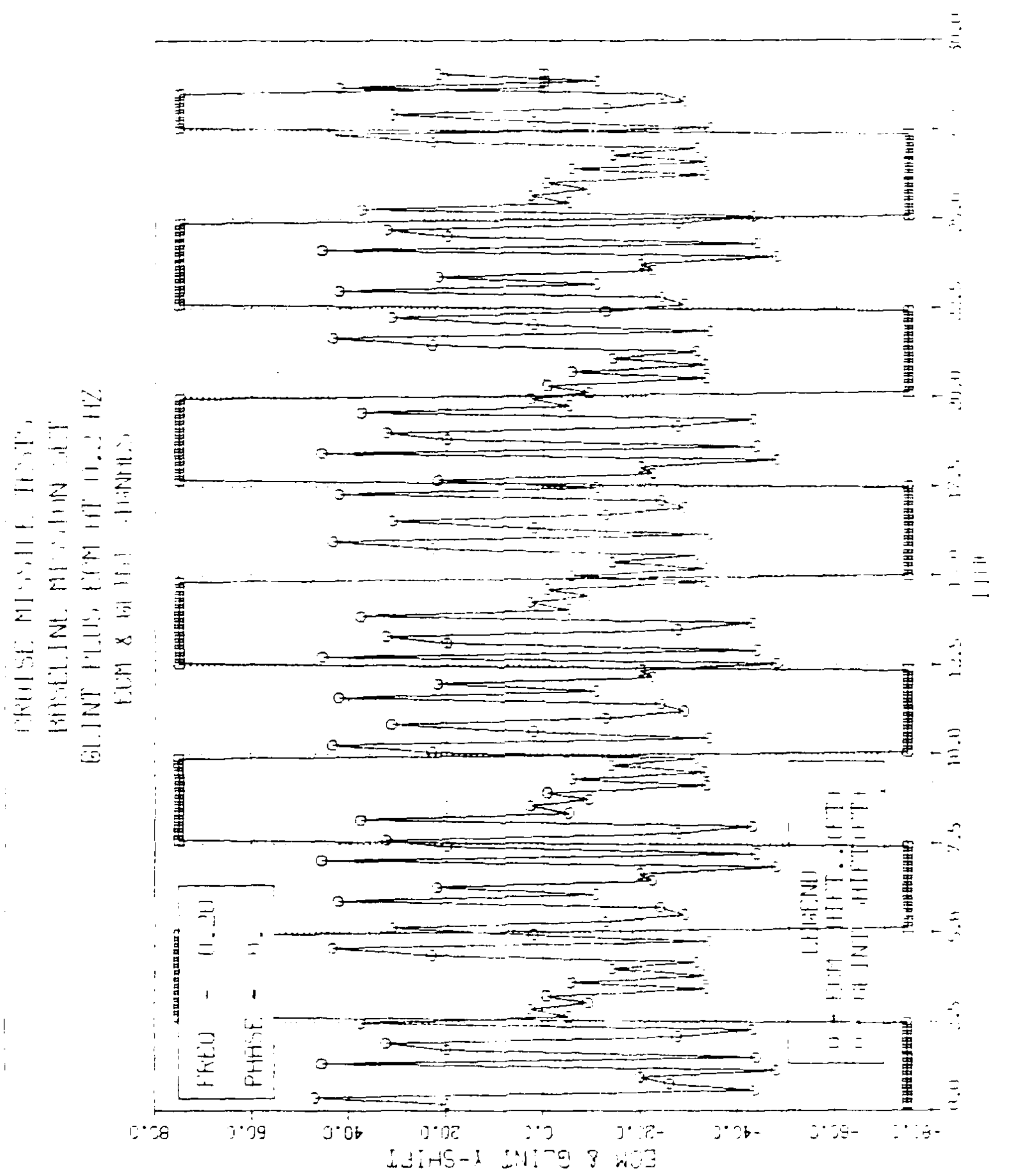


Figure A.33   Baseline with GLINT & ECM - ECM & GLINT.

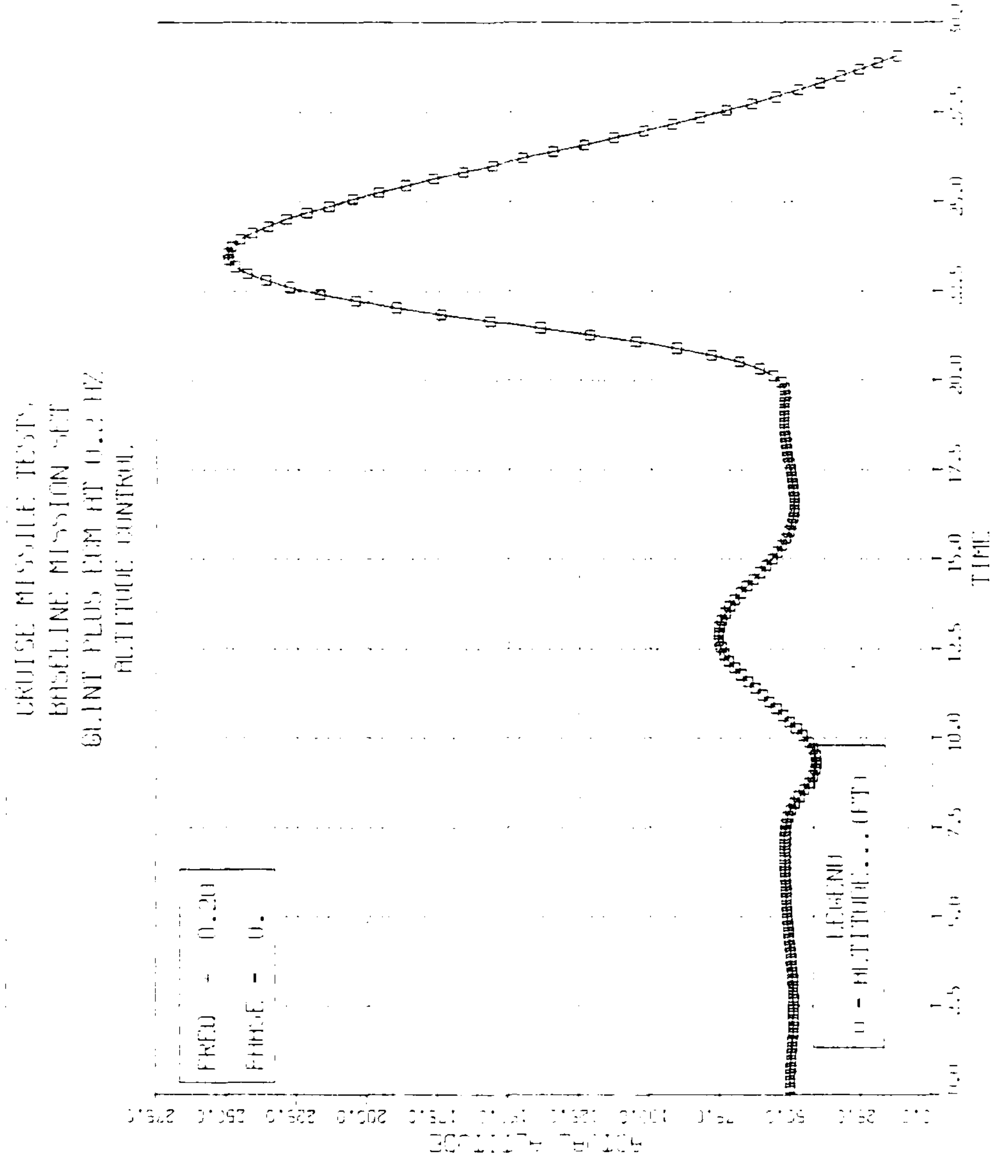


Figure A.34 Baseline with GLINT & ECM - Altitude.

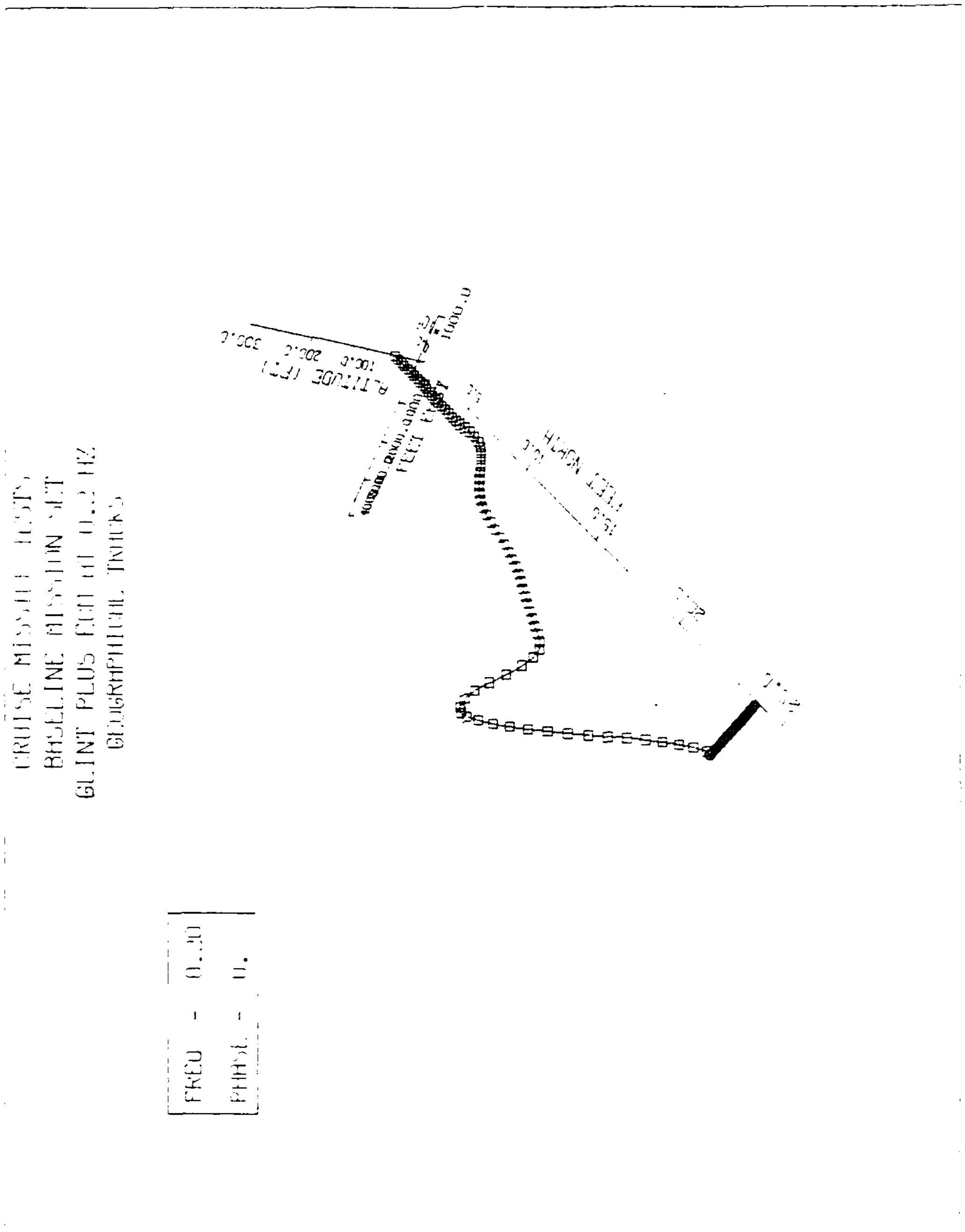


Figure A.35 Baseline with GLINT & ECM - Geo Plot.

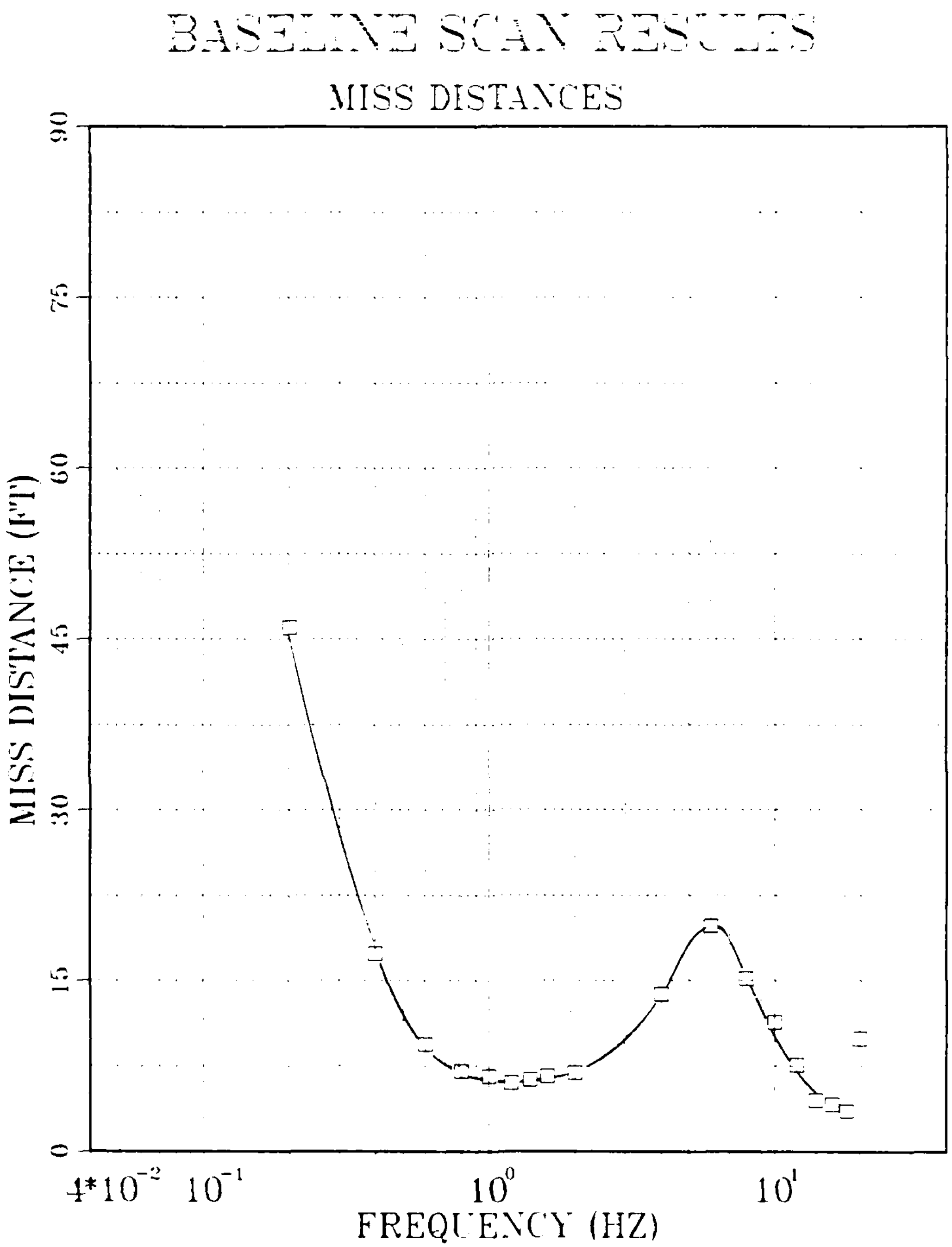


Figure A.36 Mean Miss Distances - Baseline.

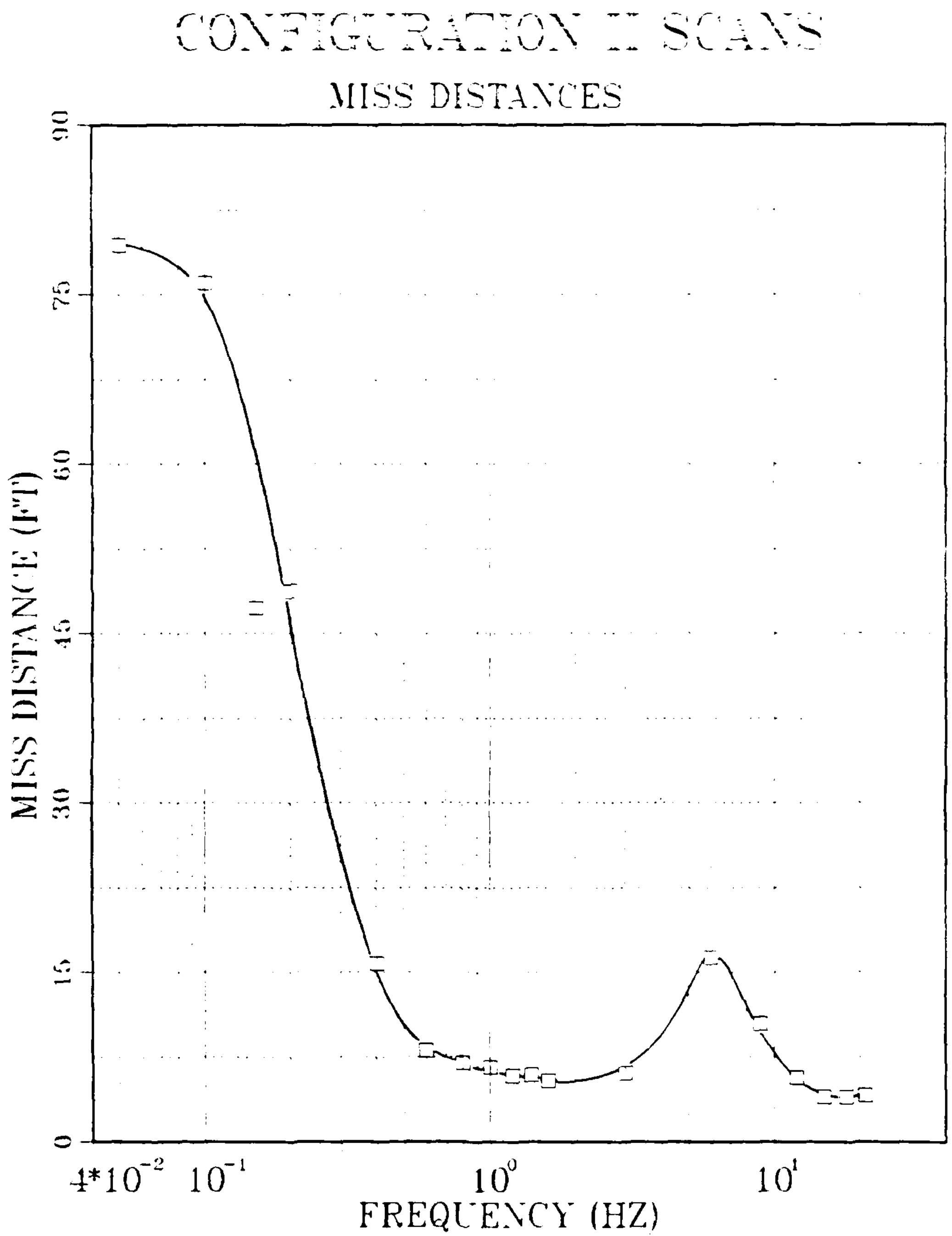


Figure A.37 Mean Miss Distances - Configuration II.

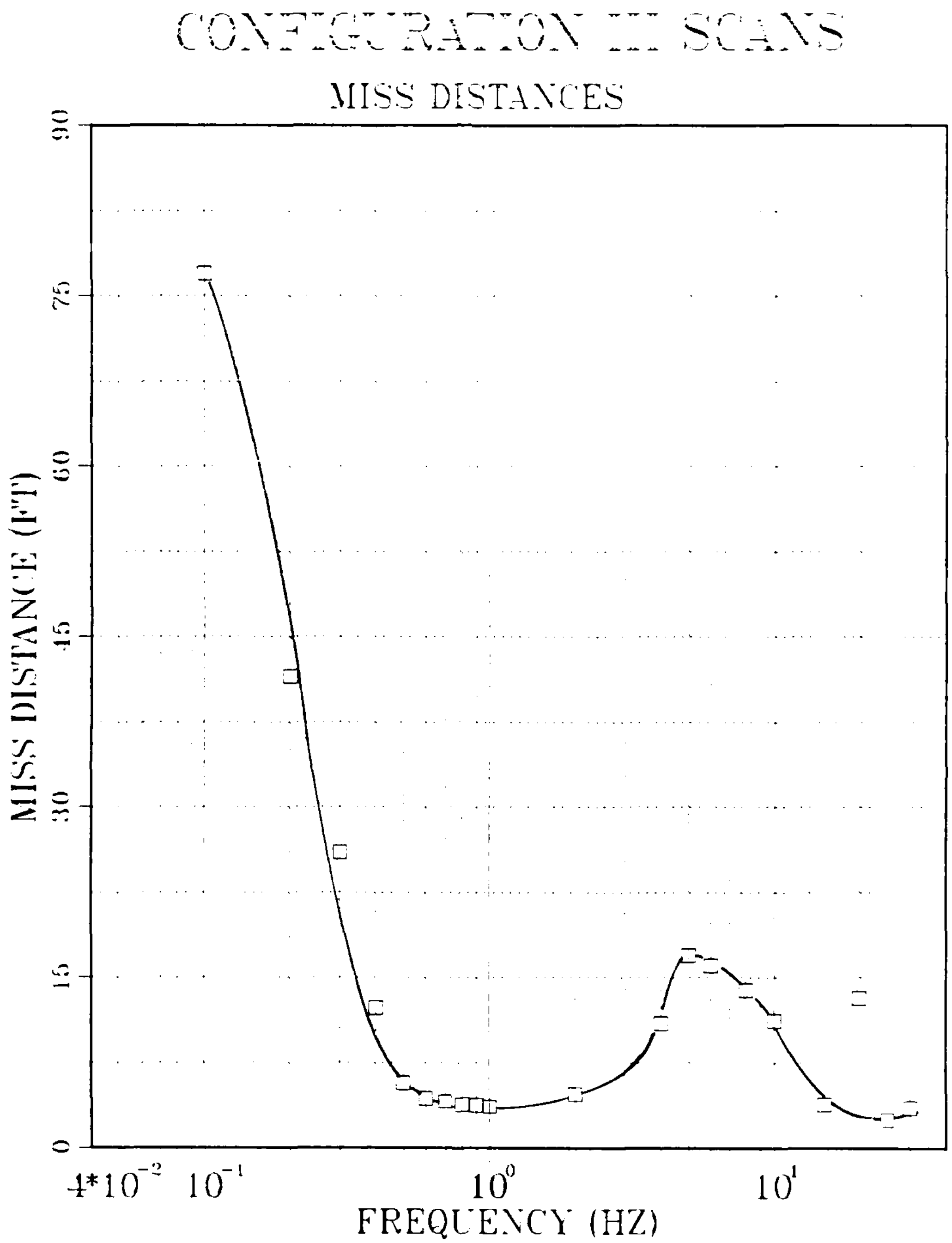


Figure A.38 Mean Miss Distances - Configuration III.

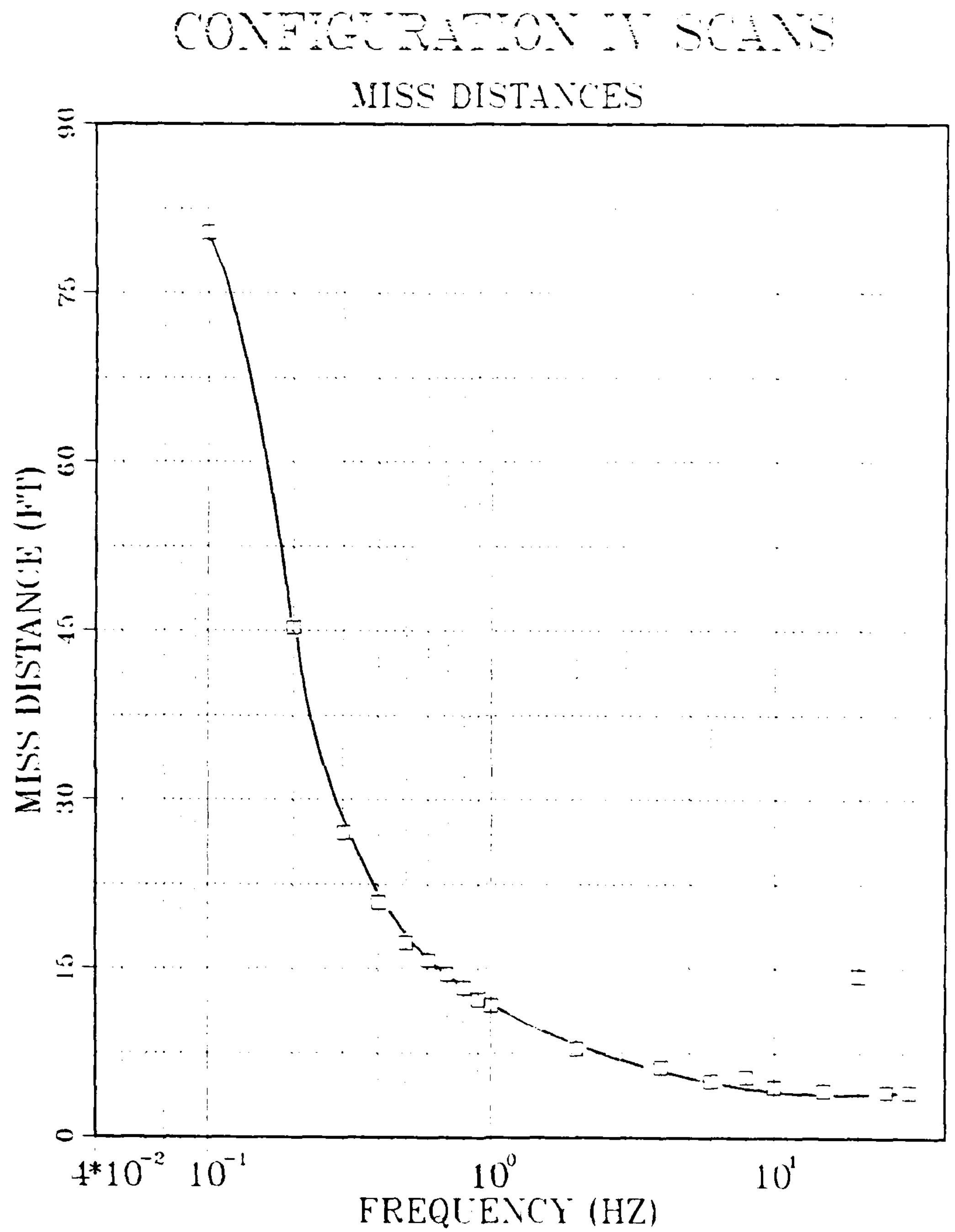


Figure A.39 Mean Miss Distances - Configuration IV.

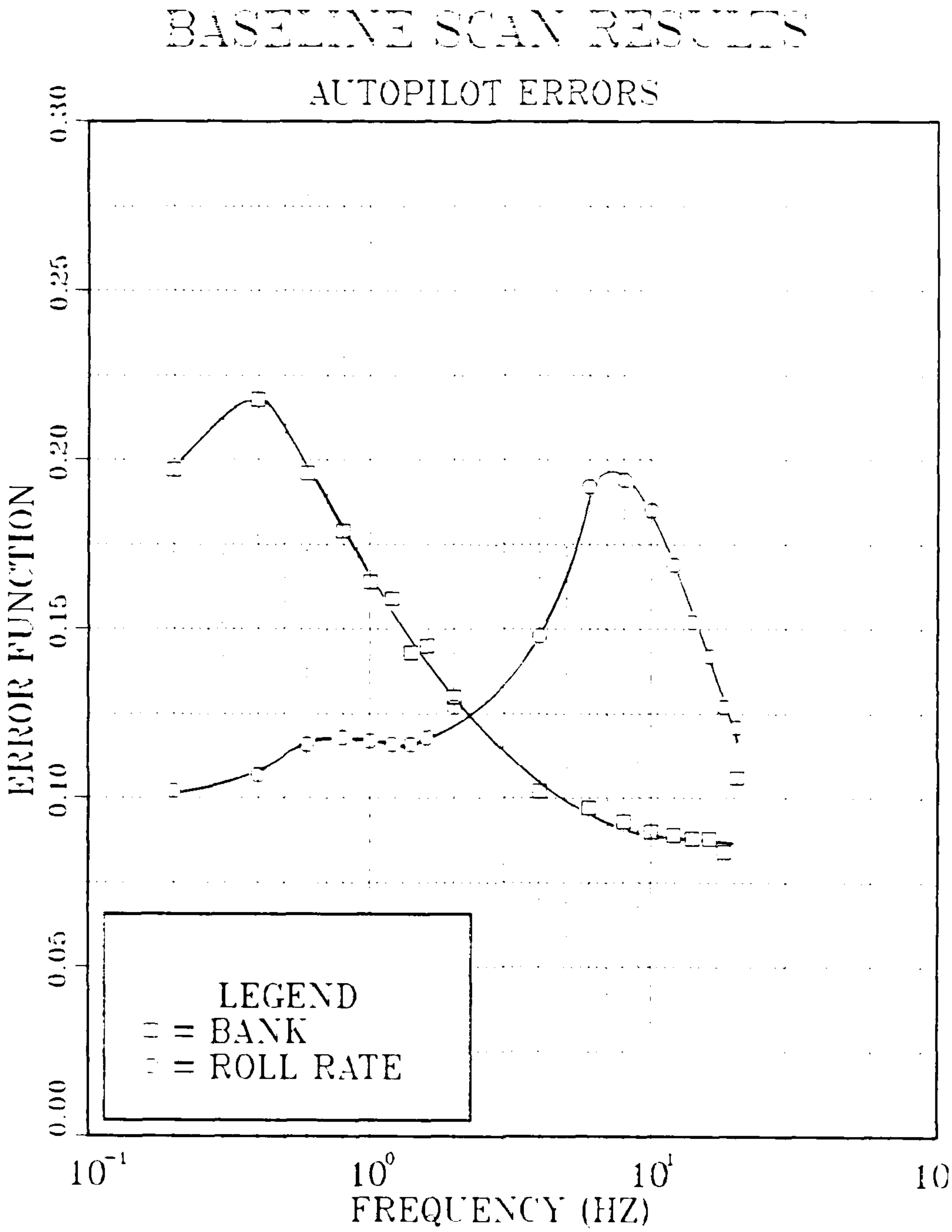


Figure A.40 Autopilot Errors - Baseline.

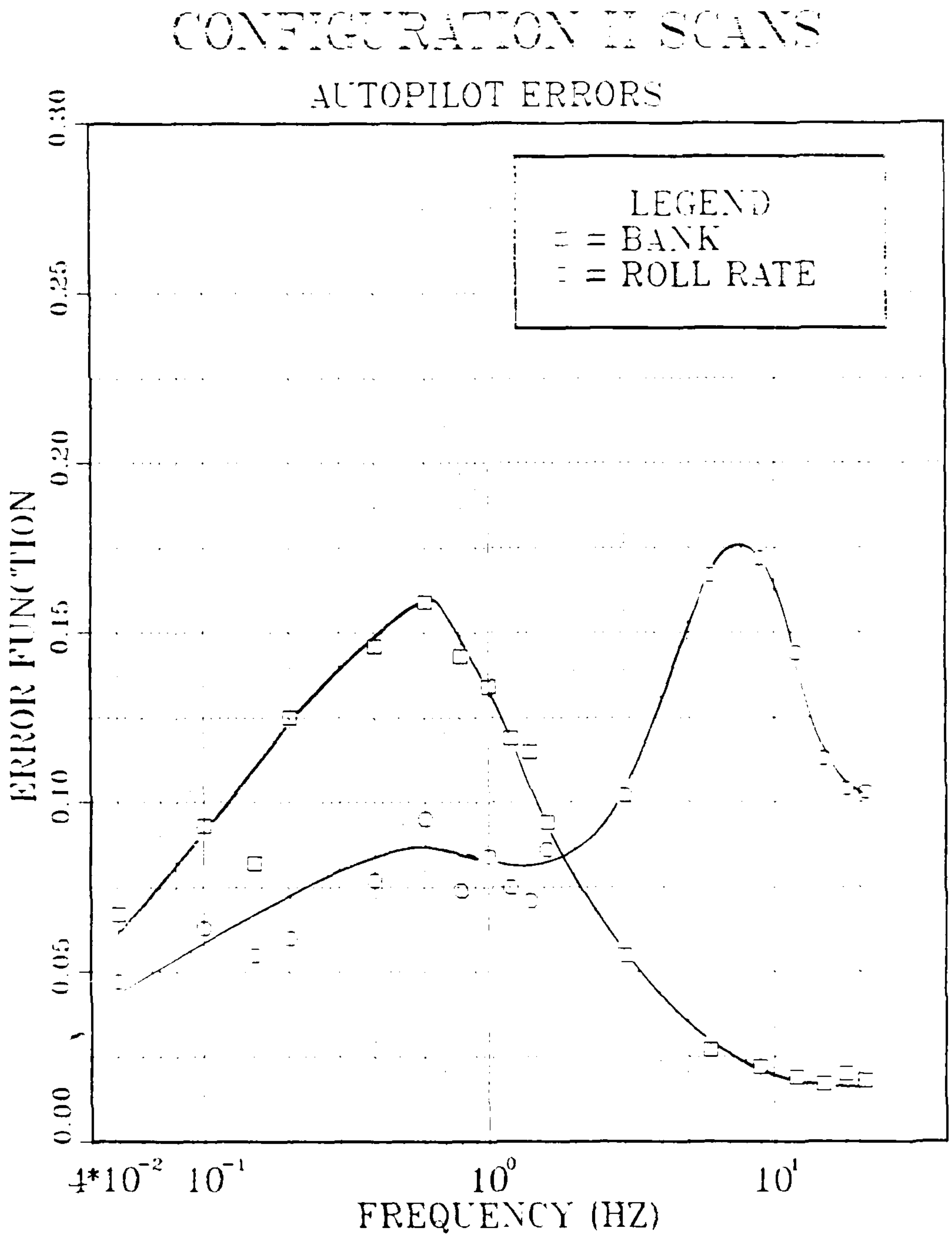


Figure A.41 Autopilot Errors - Configuration II.

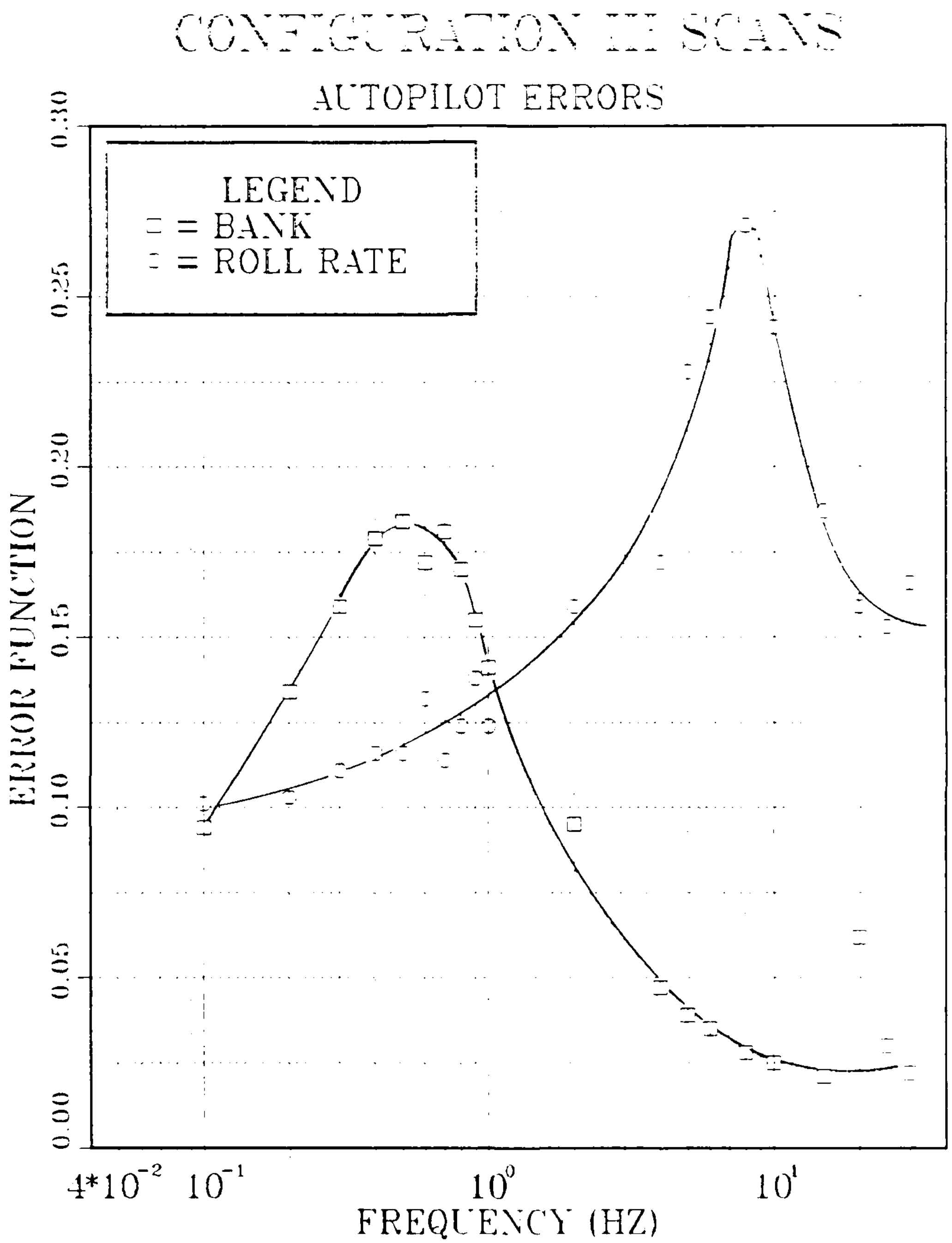


Figure A.42 Autopilot Errors - Configuration III.

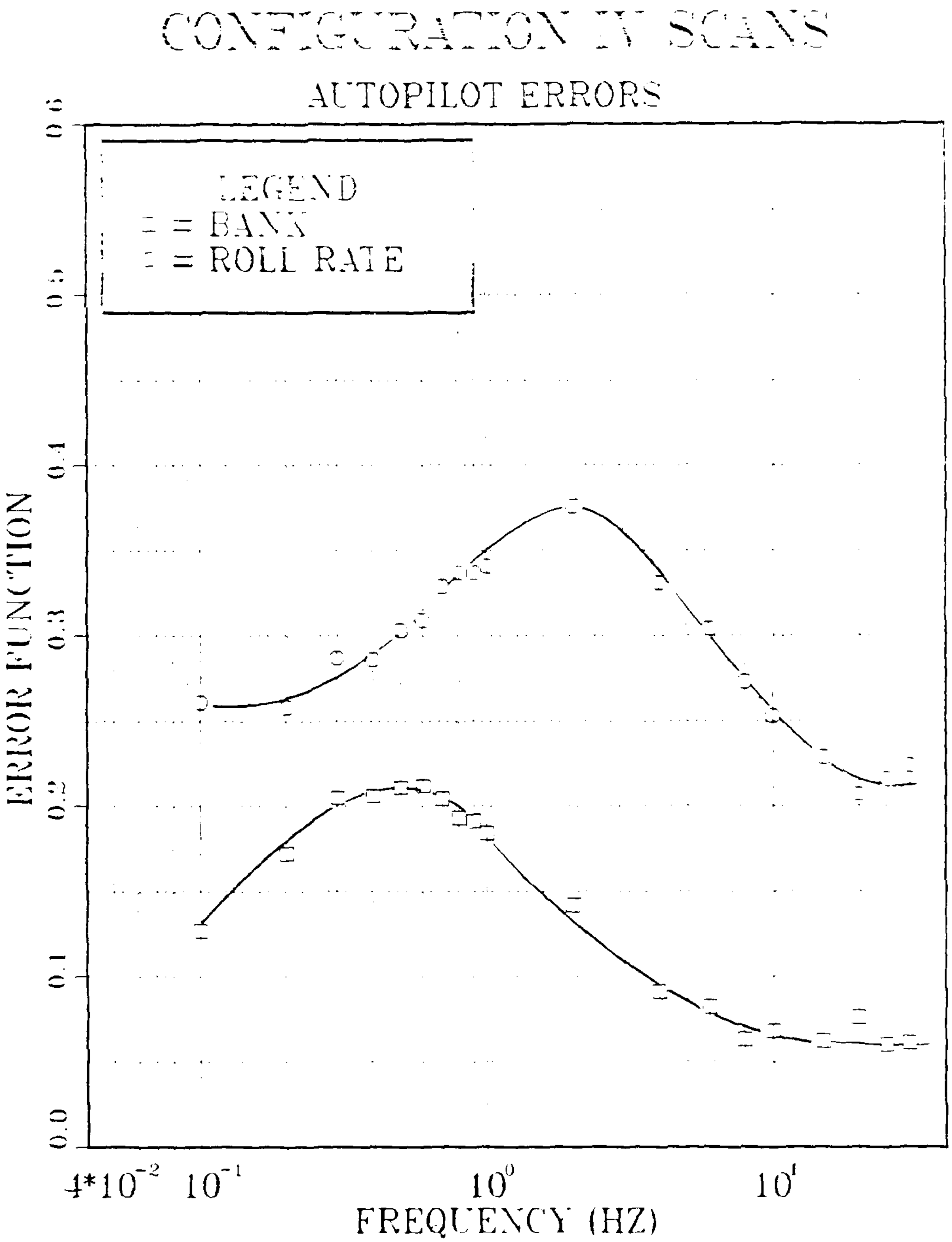


Figure A.43 Autopilot Errors - Configuration IV.

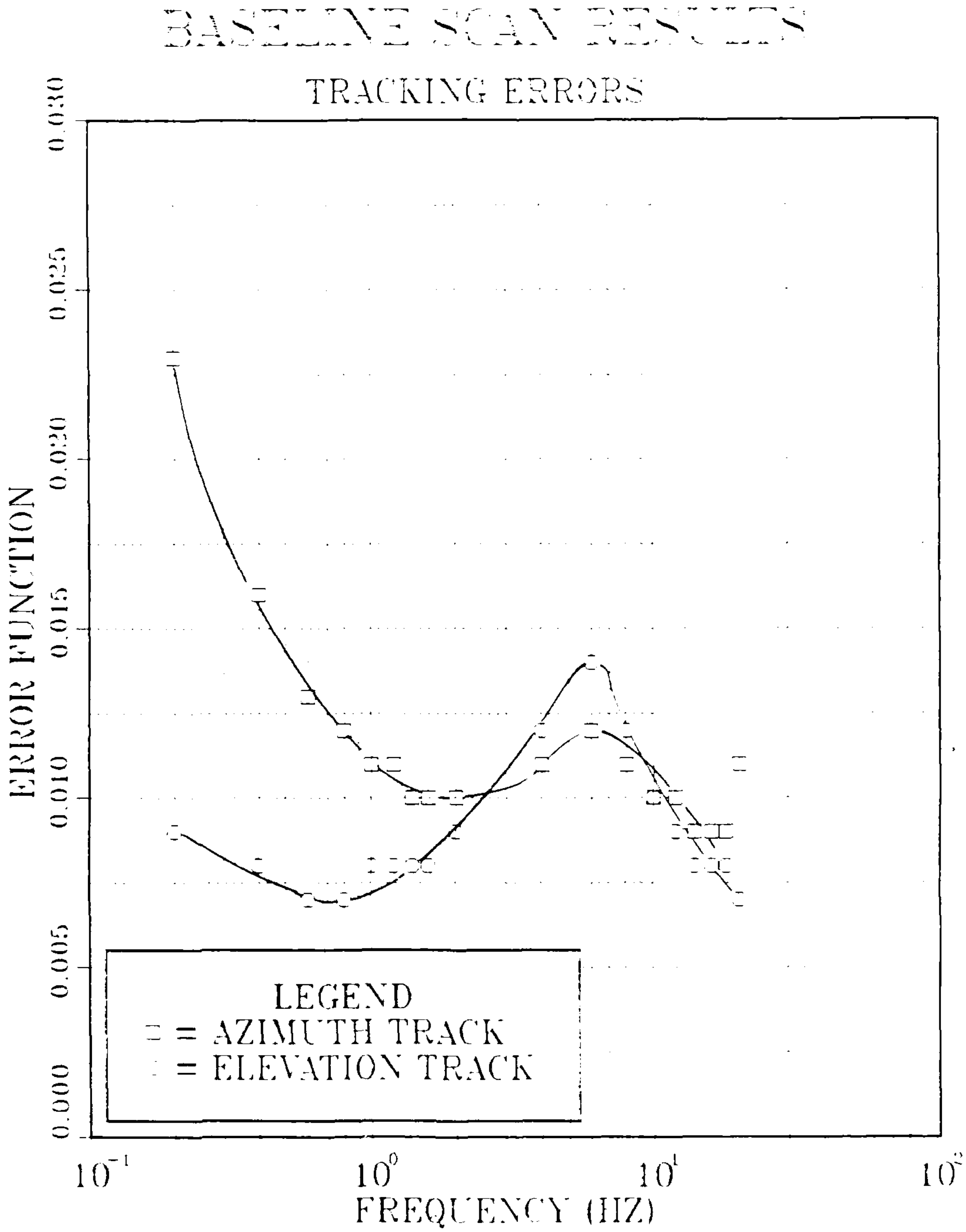


Figure A.44   Tracking Errors - Baseline.

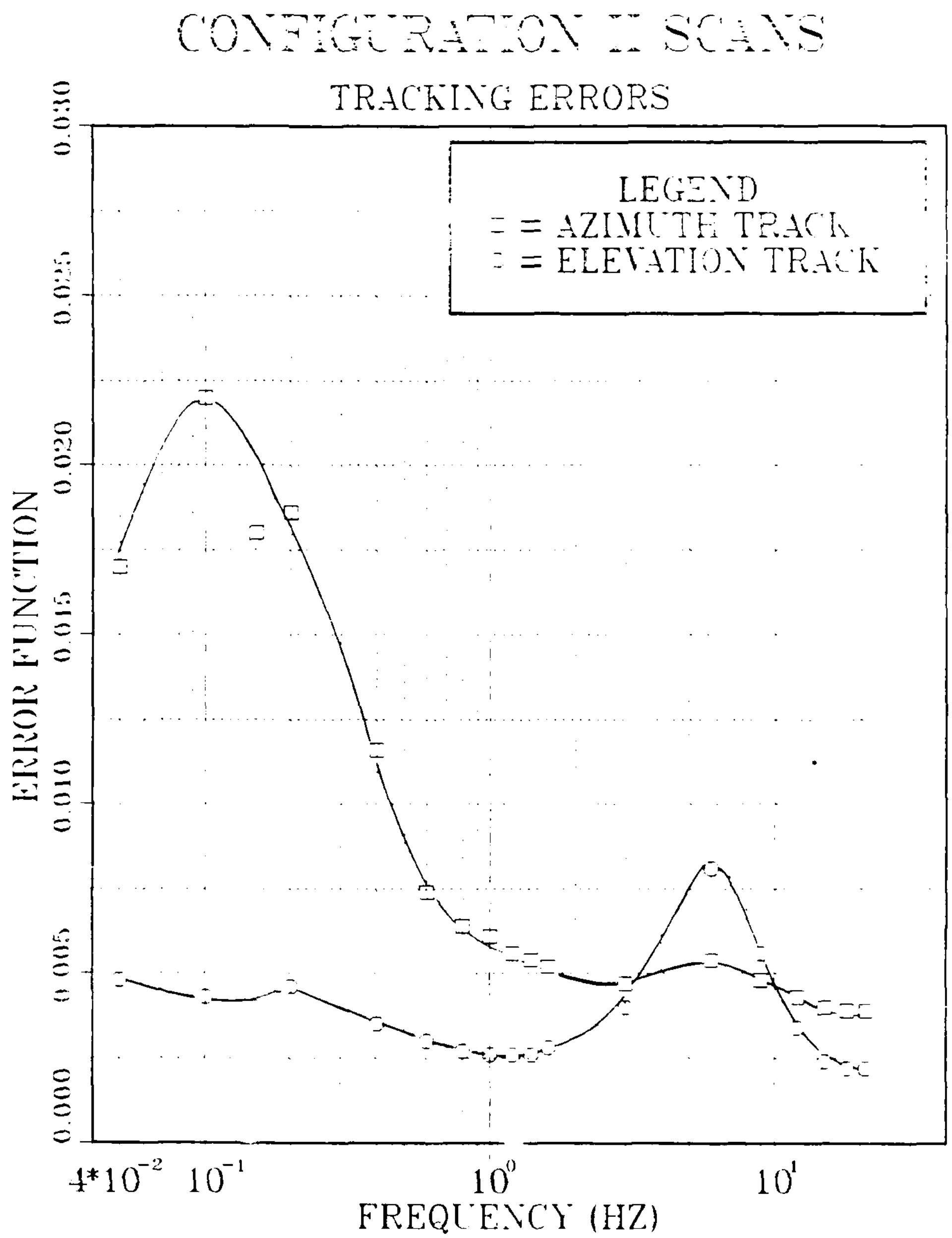


Figure A.45 Tracking Errors - Configuration II.

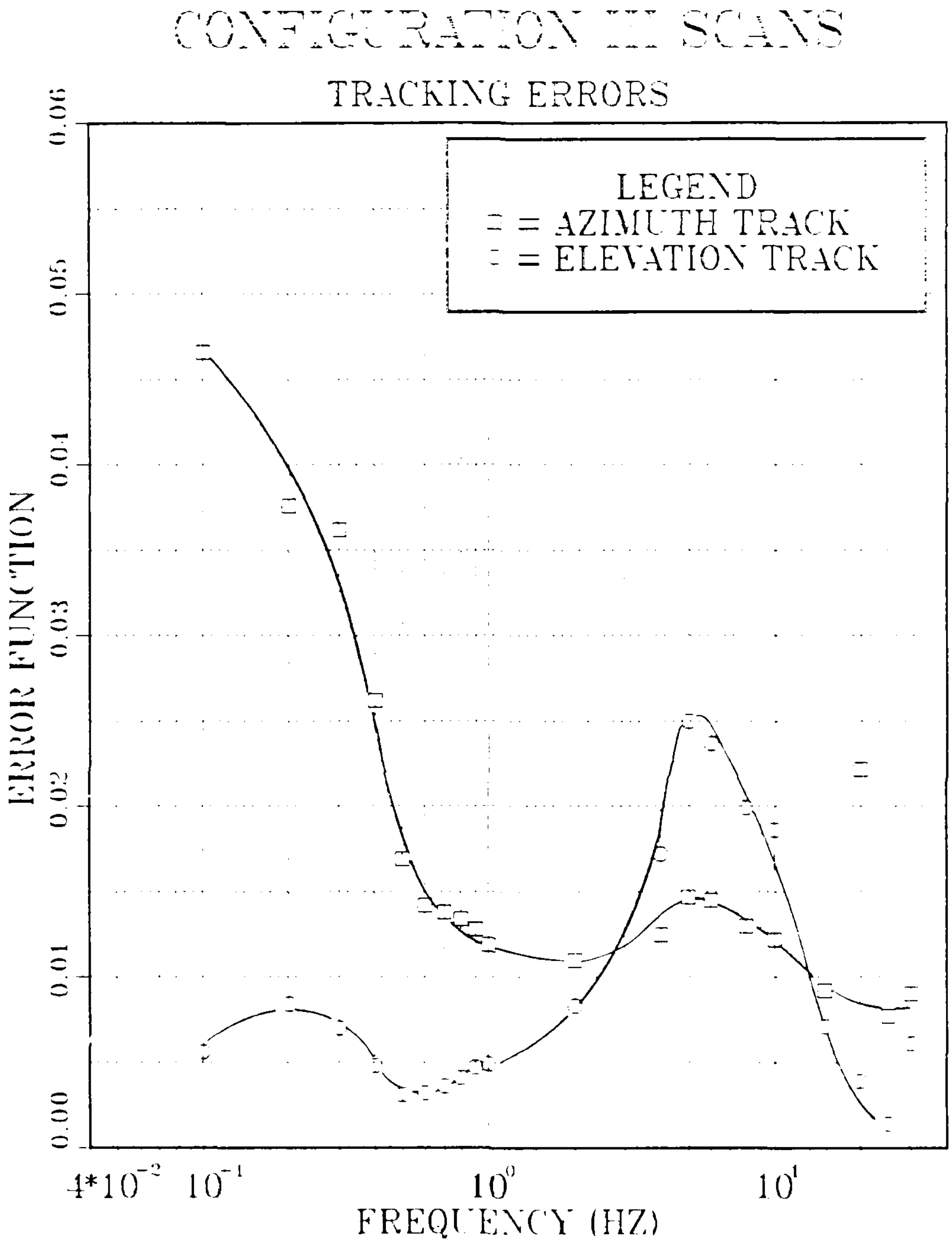


Figure A.46 Tracking Errors - Configuration III.

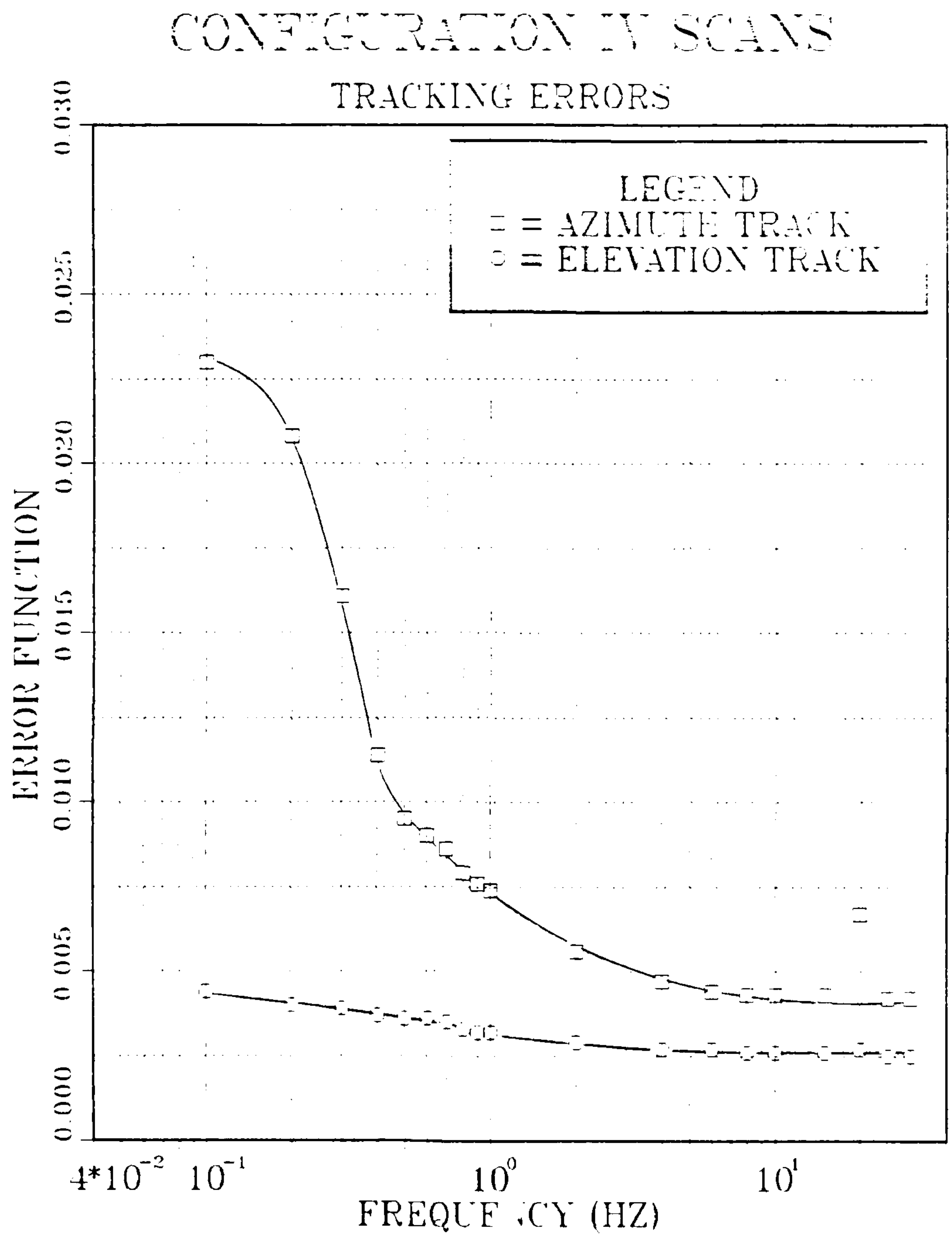


Figure A.47 Tracking Errors - Configuration IV.

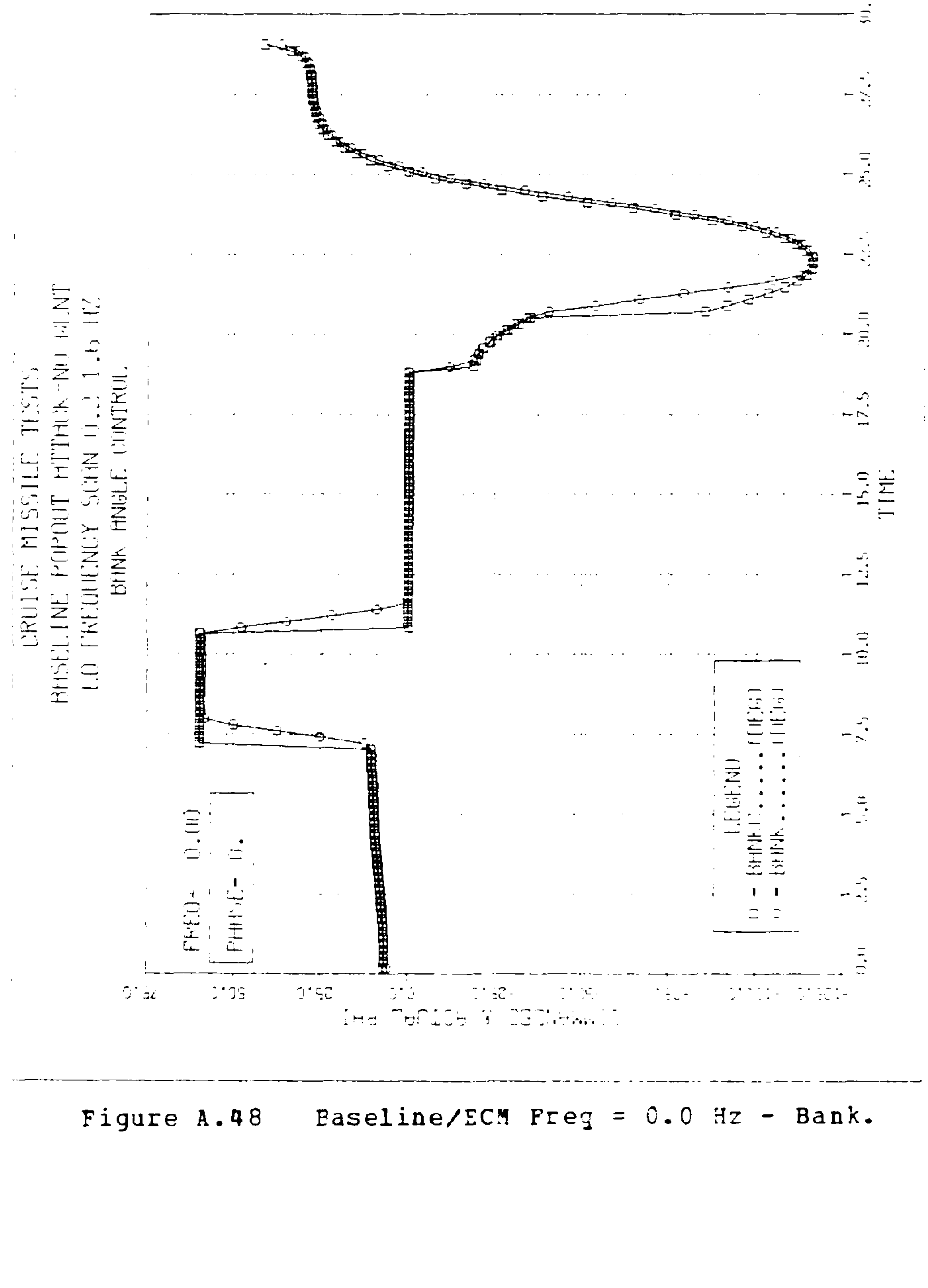


Figure A.48 Baseline/ECM Freq = 0.0 Hz - Bank.

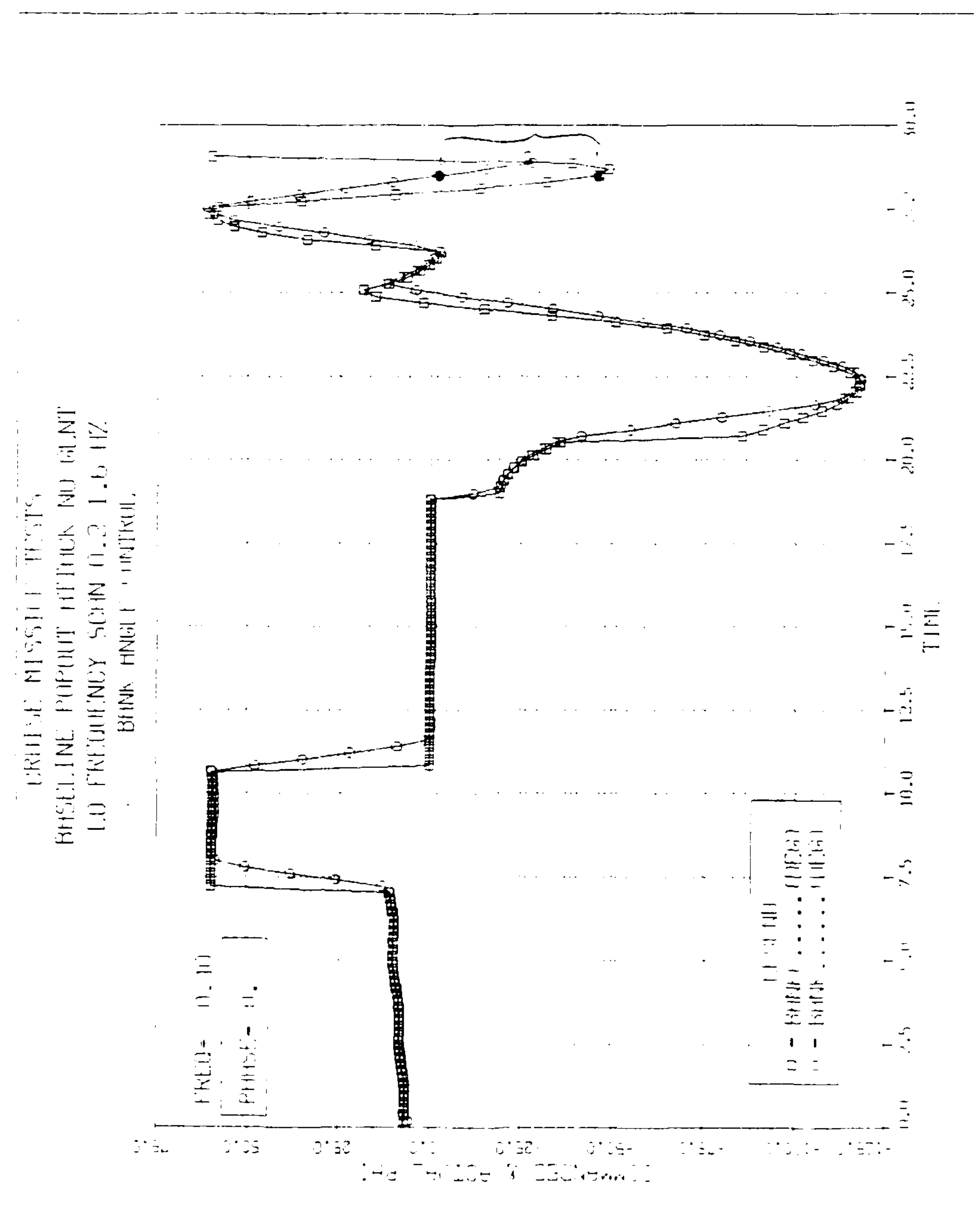


Figure A.49   Baseline/ECM Freq = 0.4 Hz - Bank.

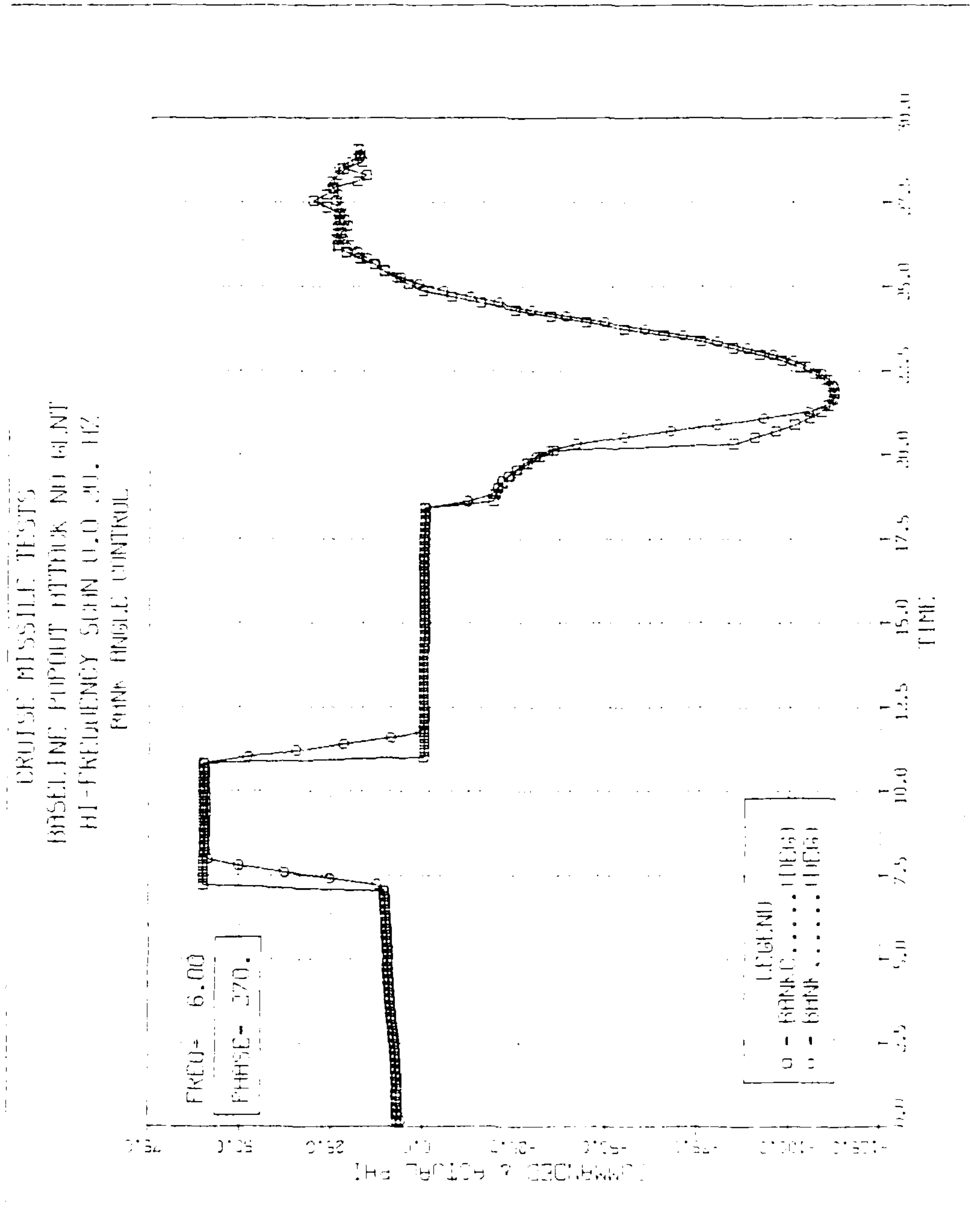


Figure A.50 Baseline/ECM Freq = 6.0 Hz - Bank.

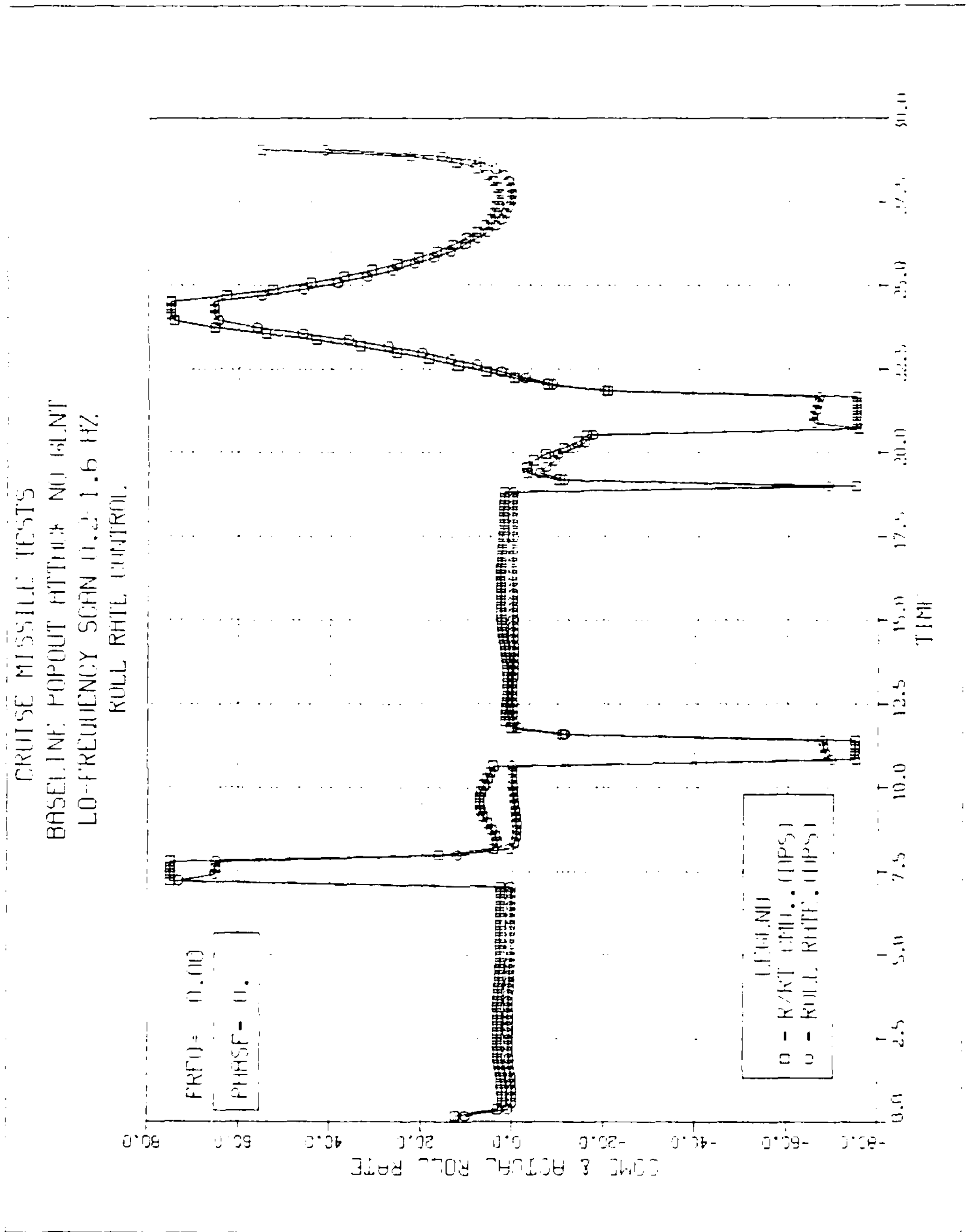


Figure A.51   Baseline/ECM Freq = 0.0 Hz - Roll Fate.

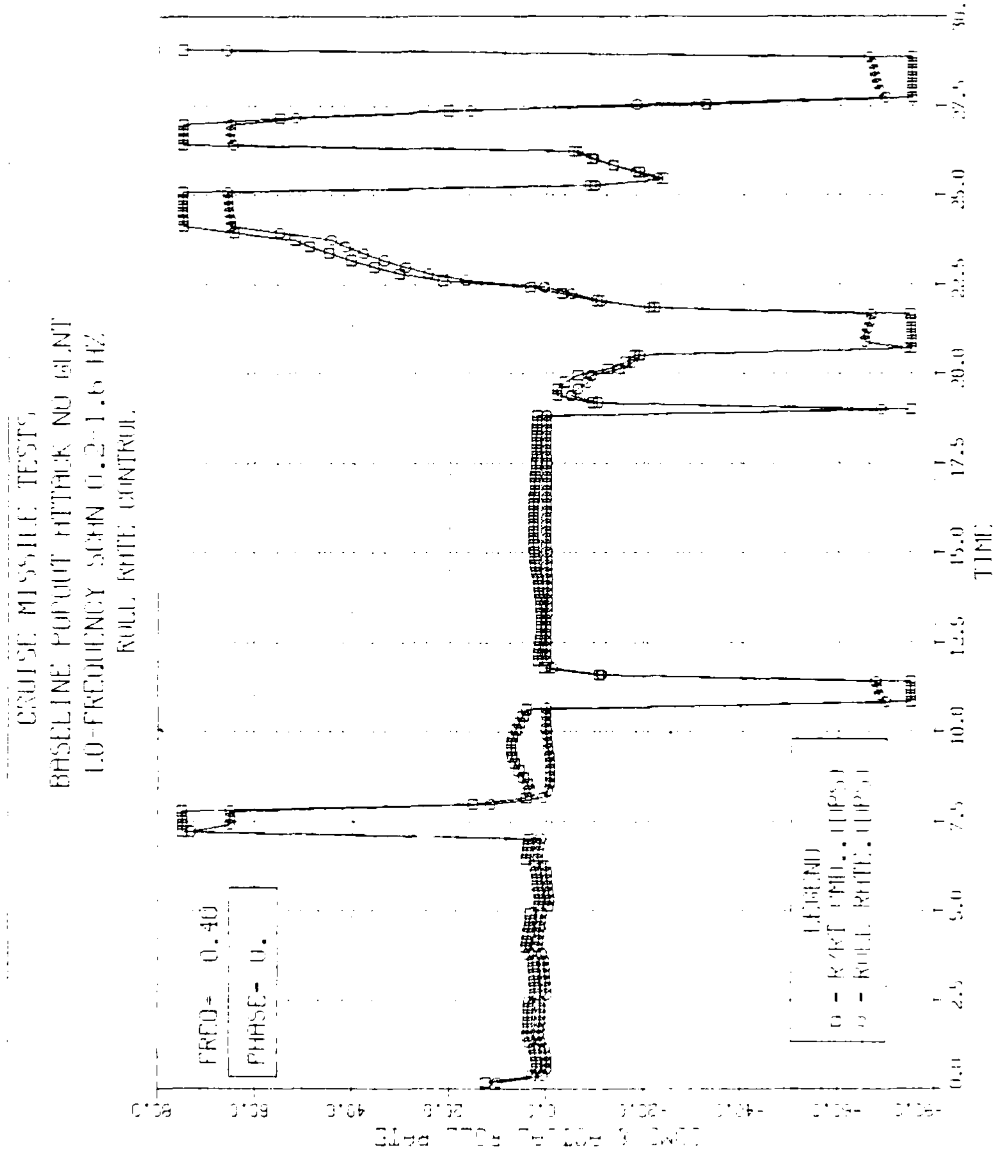


Figure A.52   Baseline/ECM Freq = 0.4 Hz - Roll Rate.

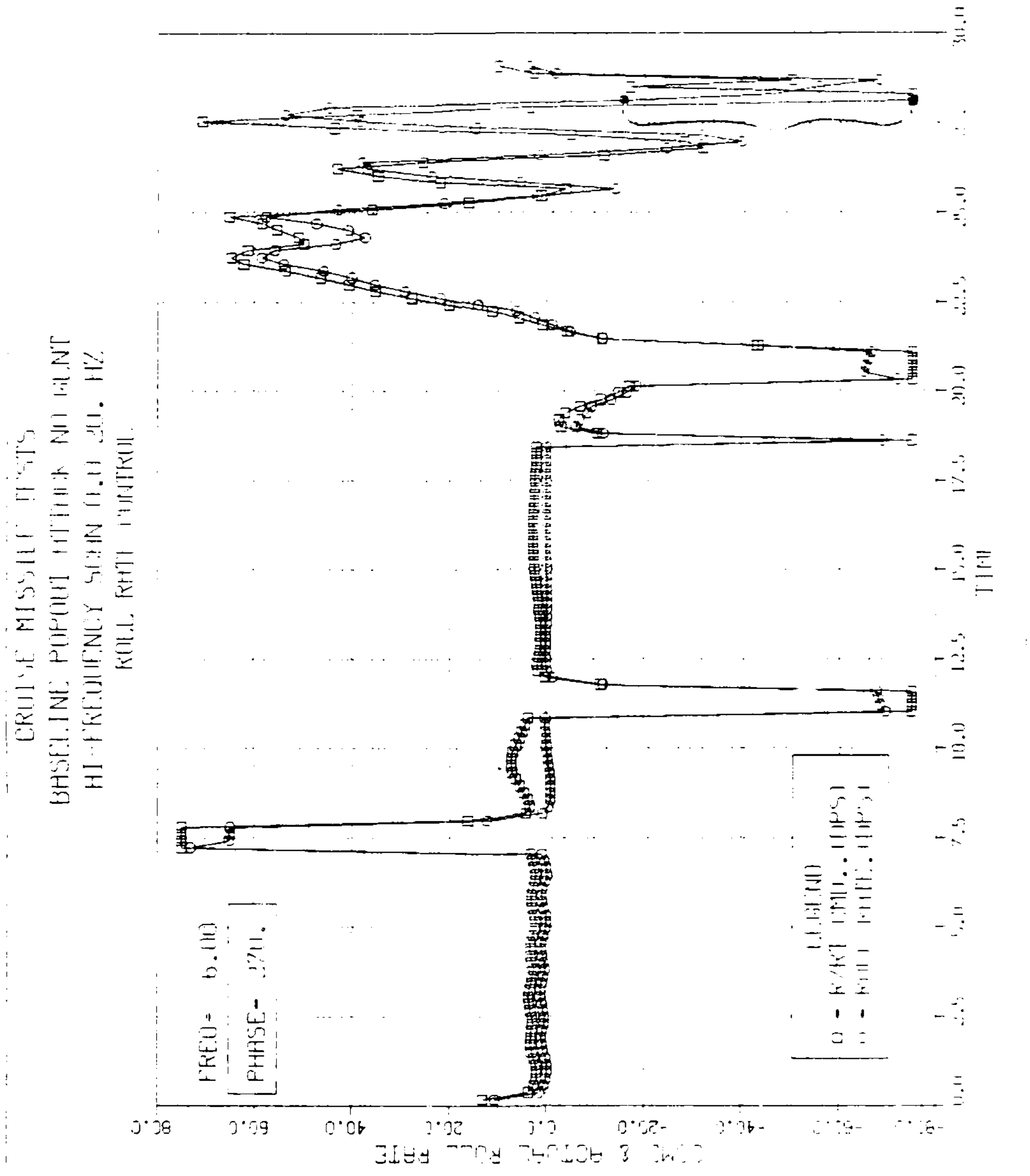


Figure A.53 Baseline/ECM Freq = 6.0 Hz - Roll Rate.

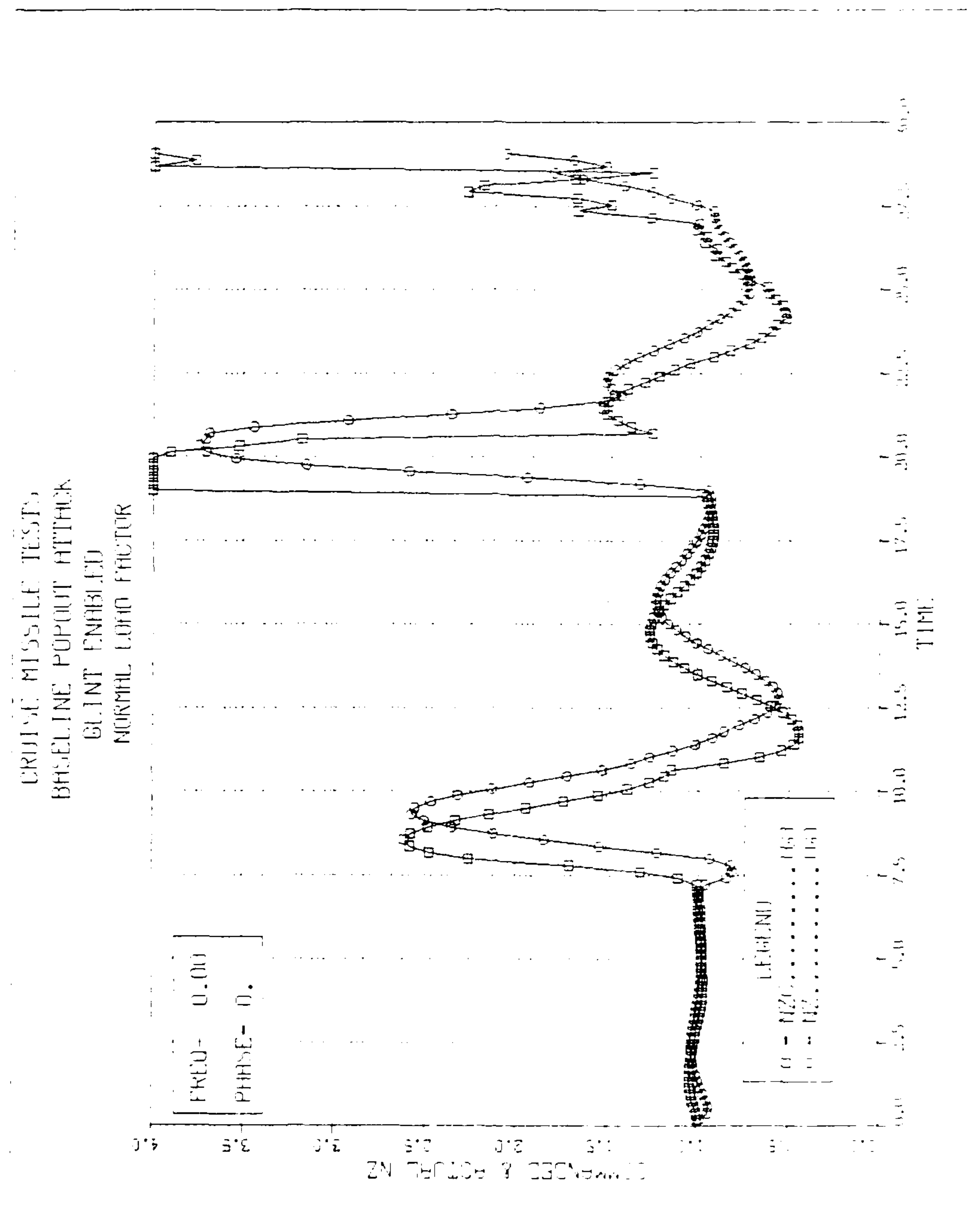


Figure A.54 Baseline with GLINT only - Load Factor.

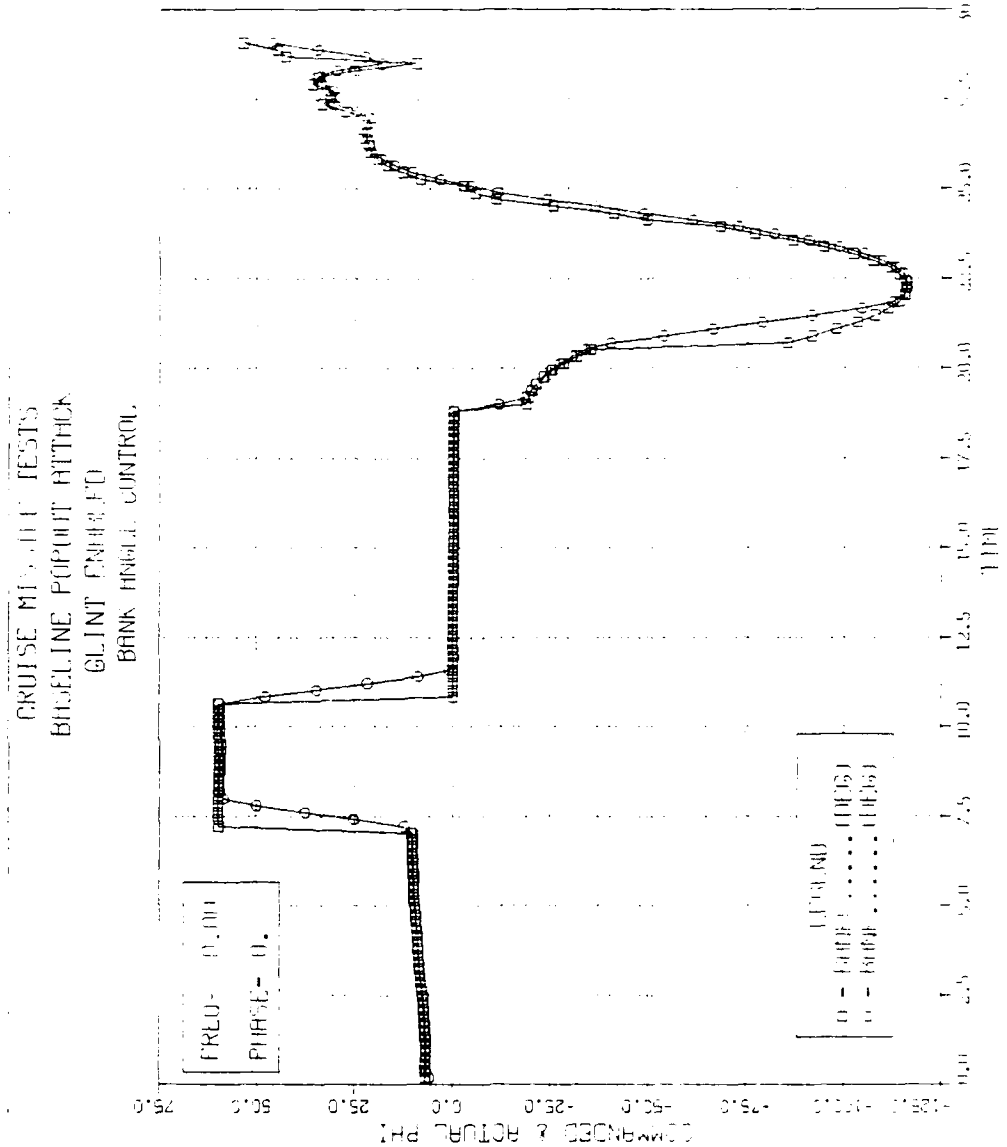


Figure A.55 Easeline with GLINT only - Bank.

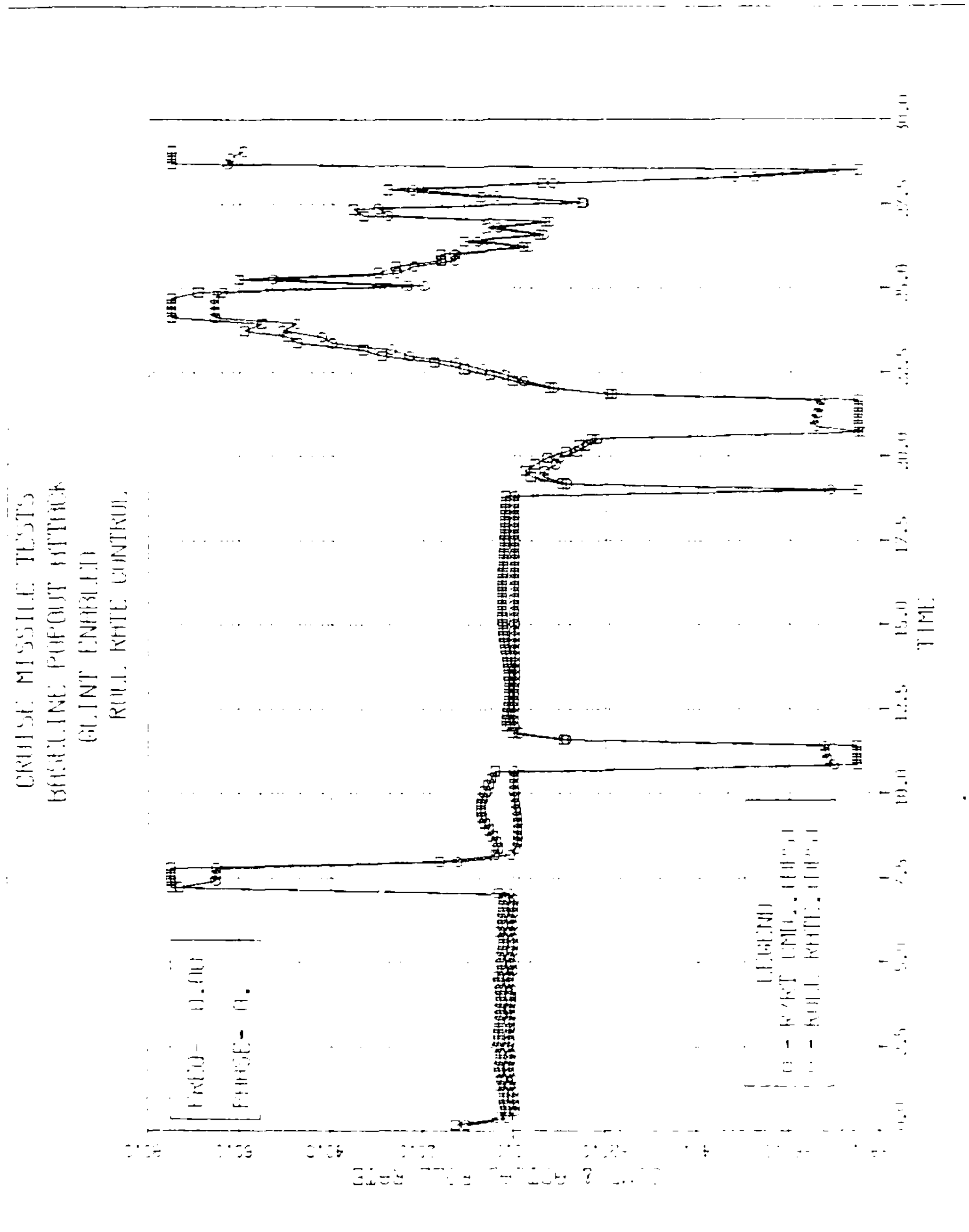


Figure A.56 Baseline with GLINT only - Roll Rate.

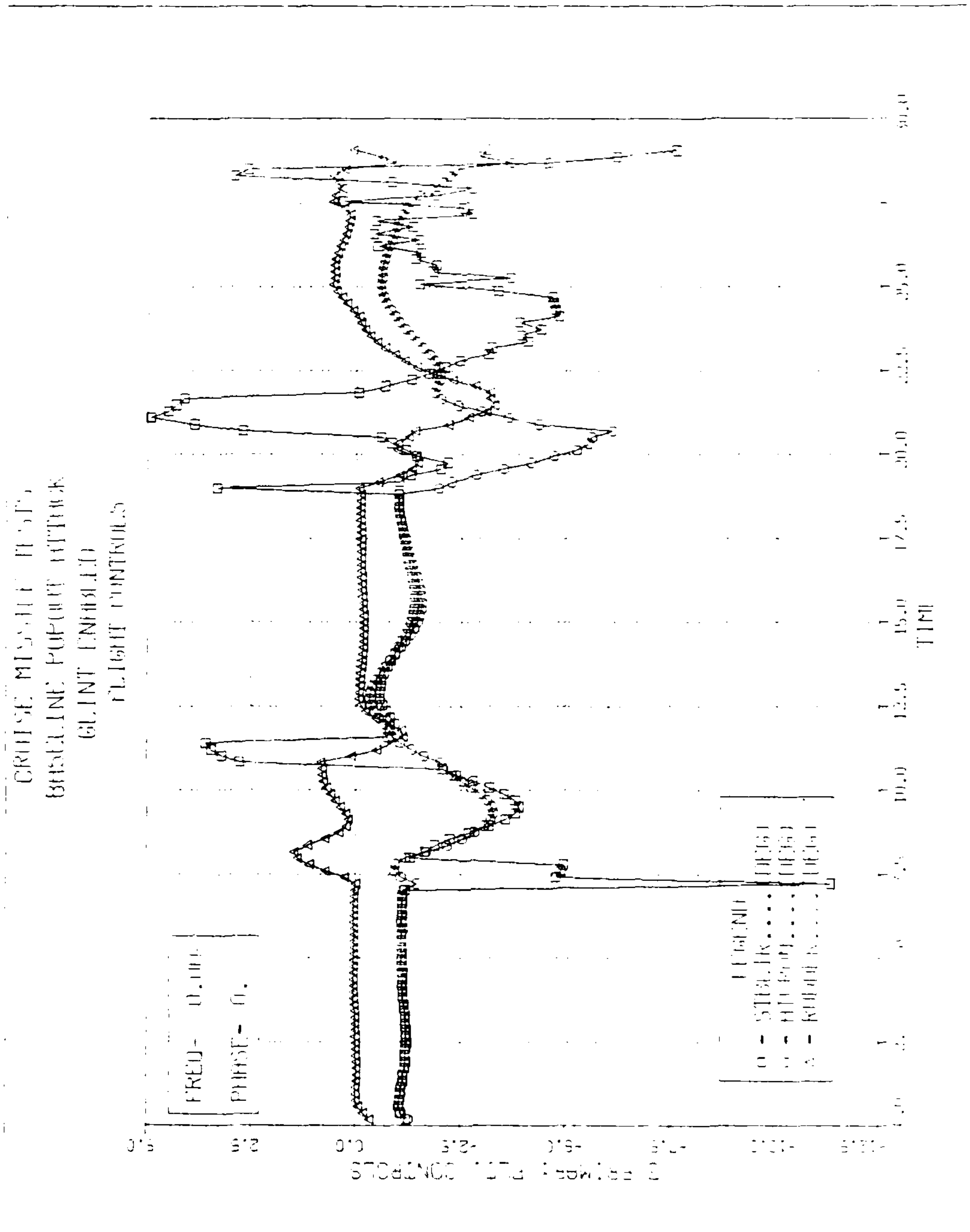


Figure A.57 Baseline with GLINT only - Controls.

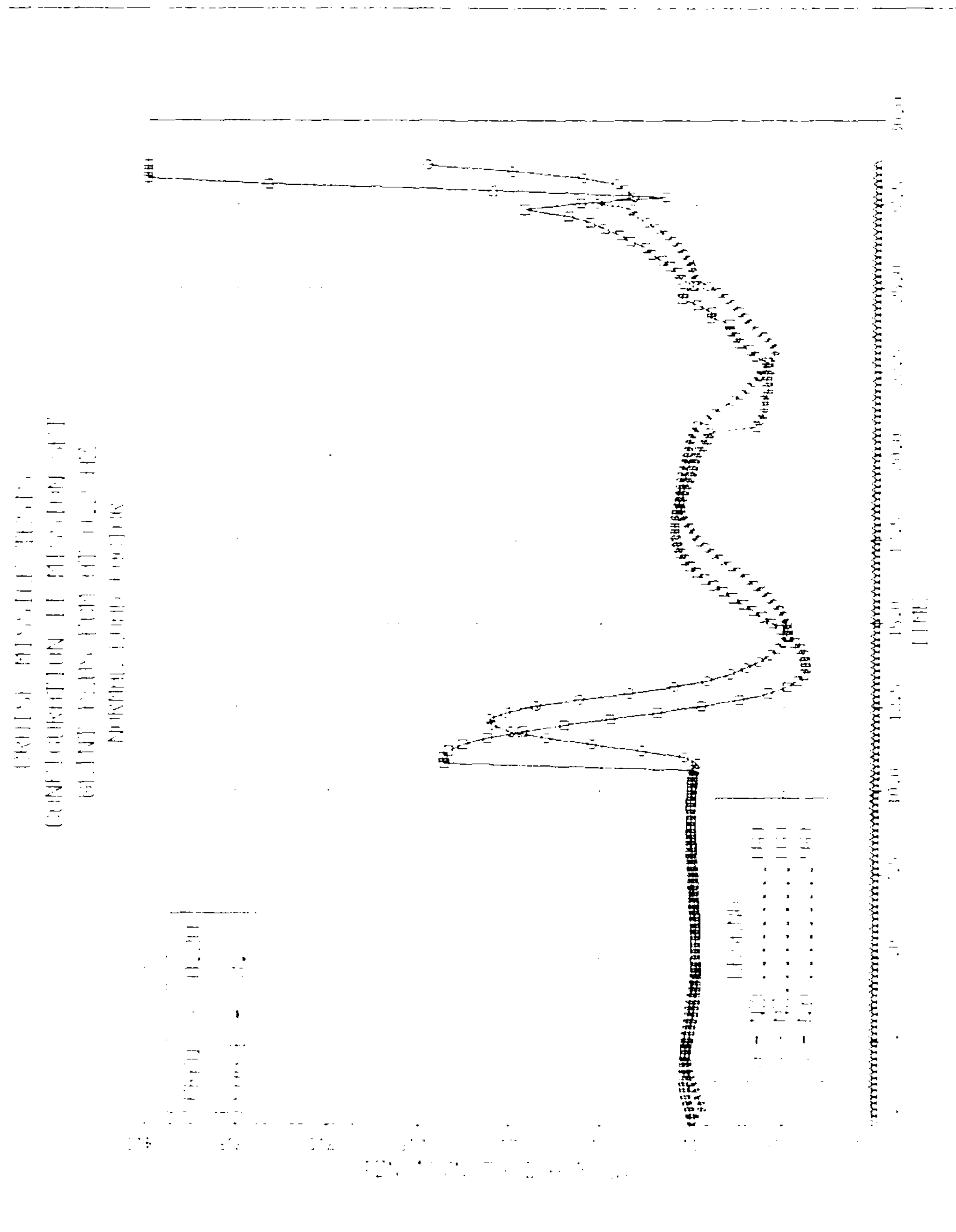
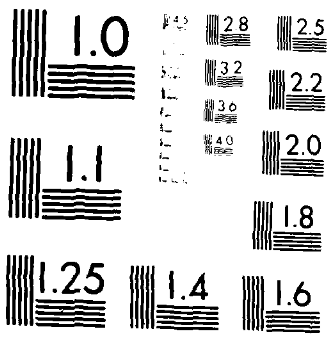


Figure A.58 Conf. II Mission Set - Load Factor.

AD-A152 193 AN INVESTIGATION INTO THE CONTROL LIMITATIONS OF A BANK TO TURN MISSILE IN THE TERMINAL HOMING PHASE(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA B P ANDERSON SEP 84 2/3

UNCLASSIFIED F/G 17/7 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

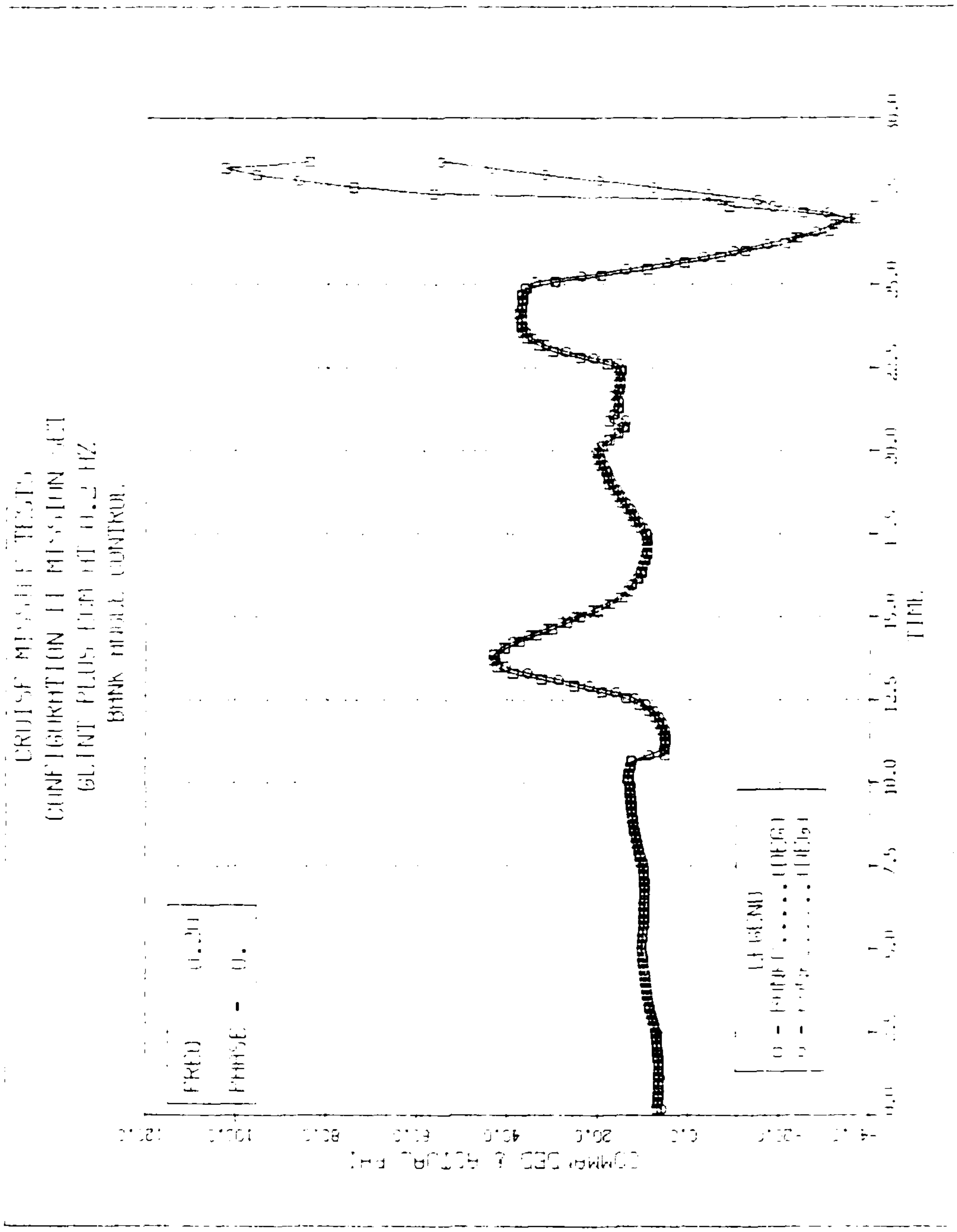


Figure A.59 Conf. II Mission Set - Bank.

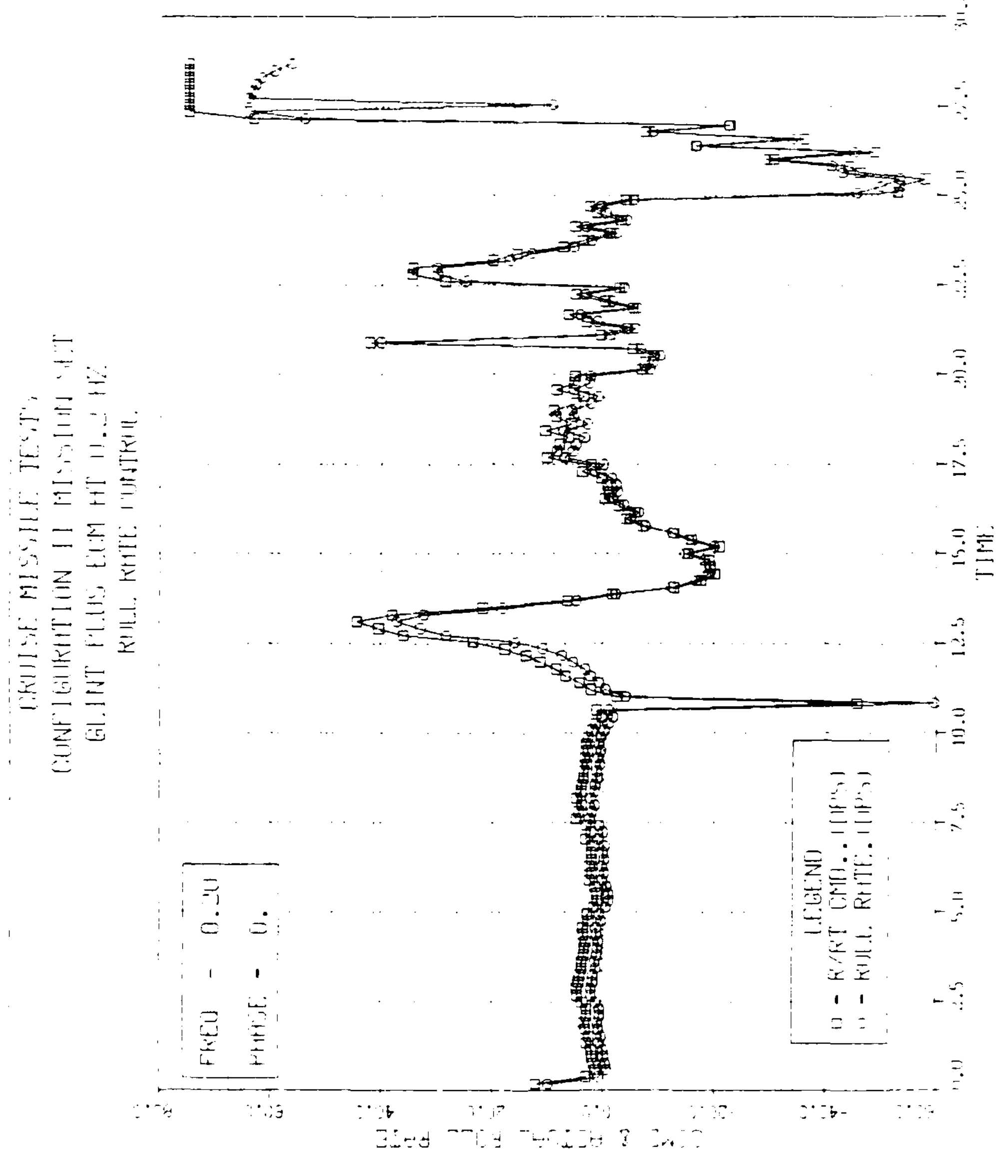


Figure A.60 Conf. II Mission Set - Roll Rate.

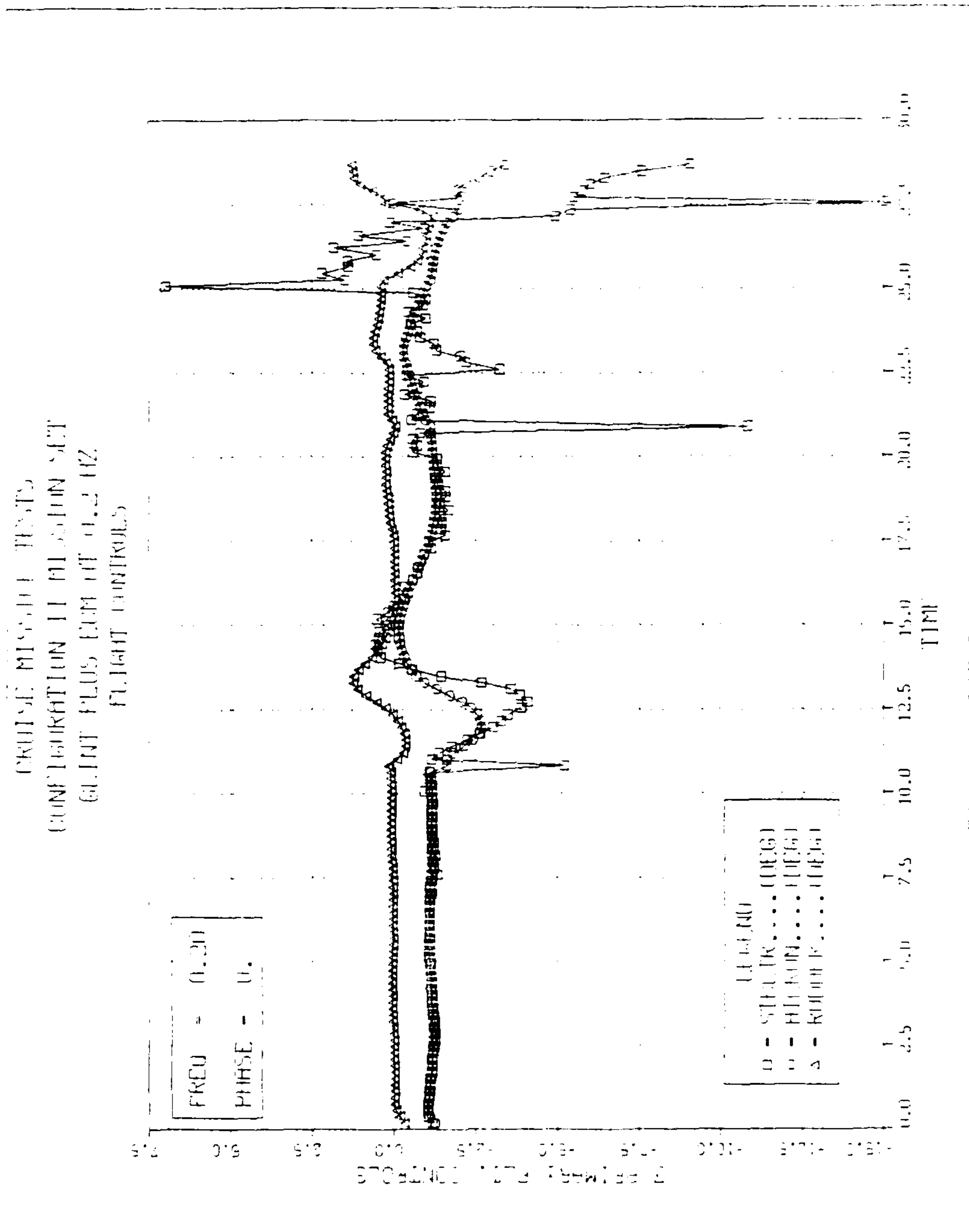


Figure A.61 Conf. II Mission Set - Controls.

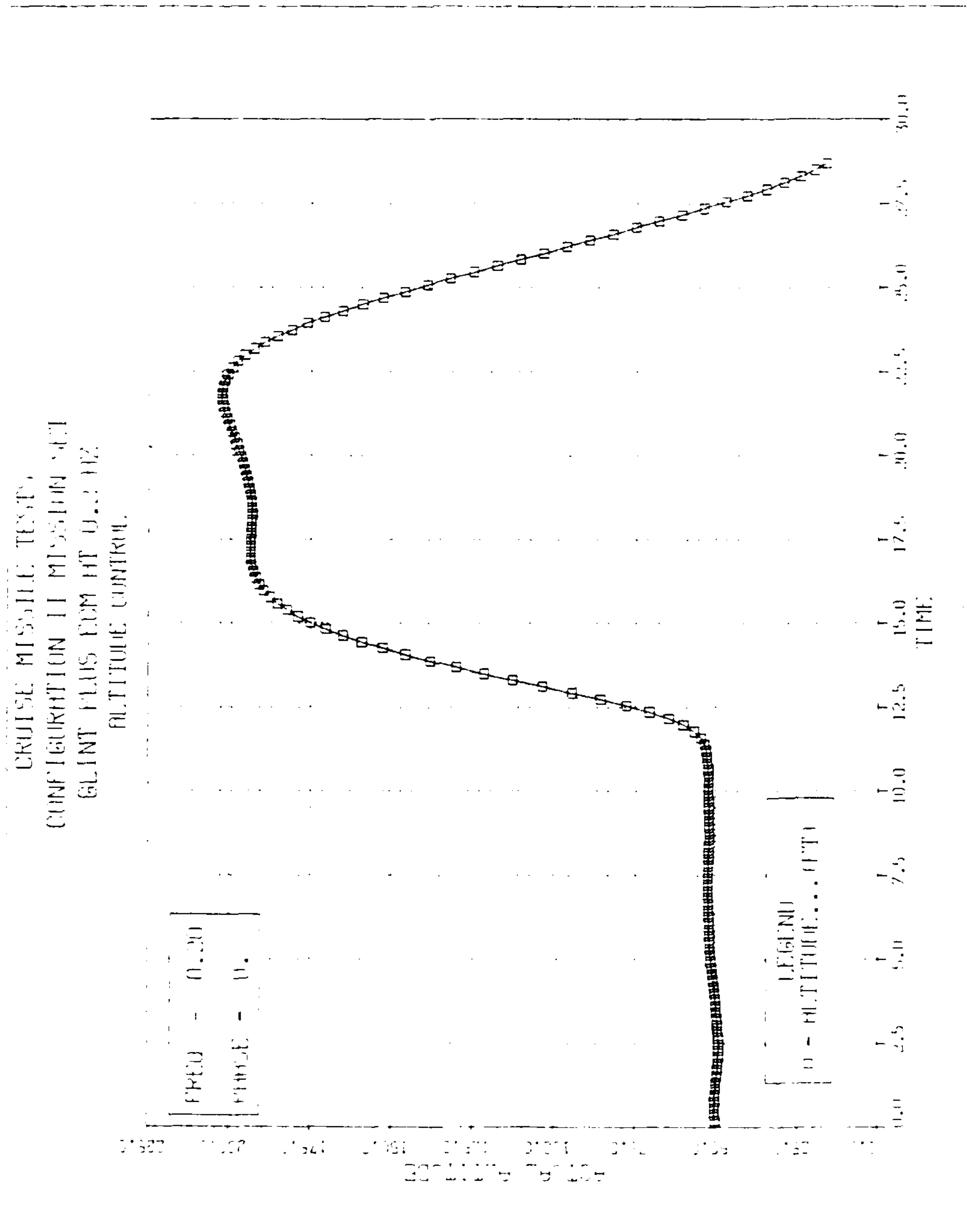


Figure A.62 Conf. II Mission Set - Altitude.

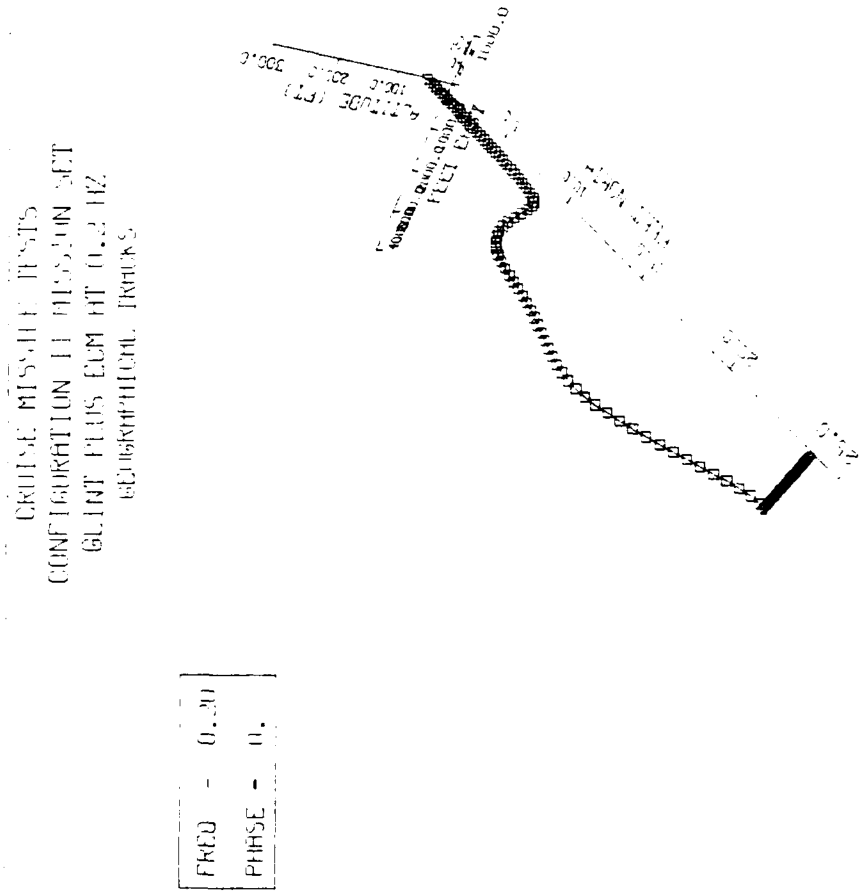


Figure A.63 Conf. II Mission Set - Geo Plot.

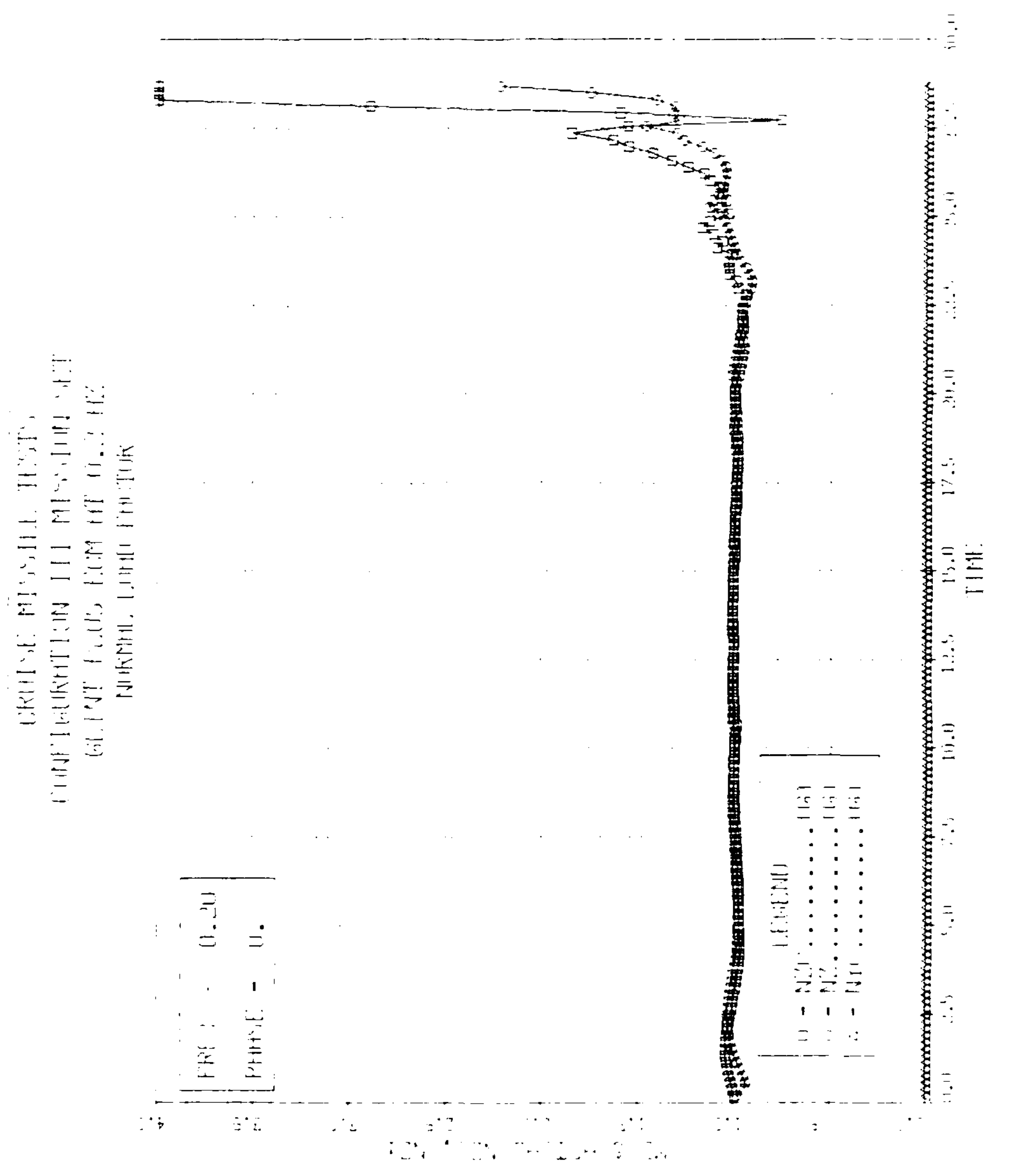


Figure A.64 Conf. III Mission Set - Load Factor.

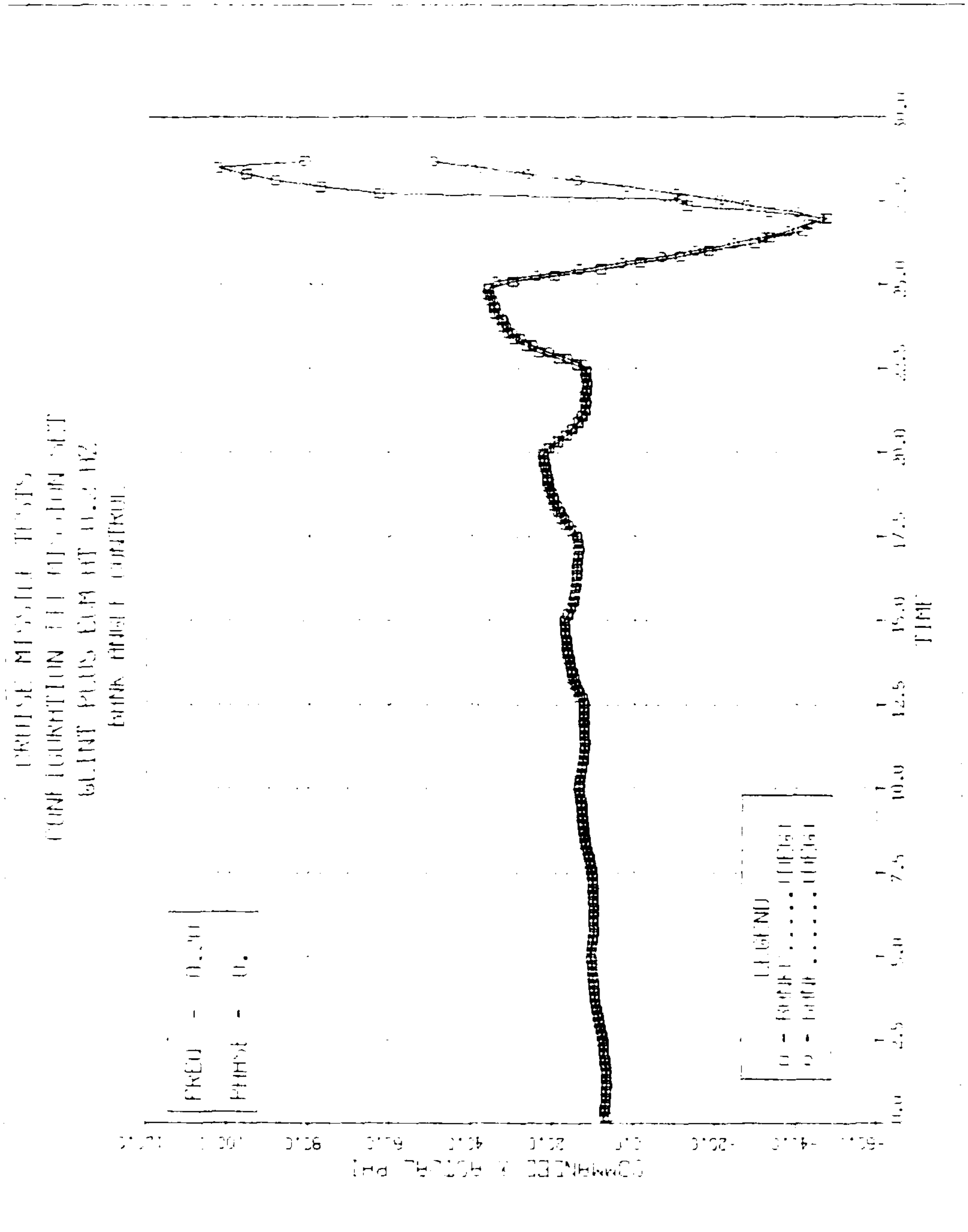


Figure A.65 Conf. III Mission Set - Bank.

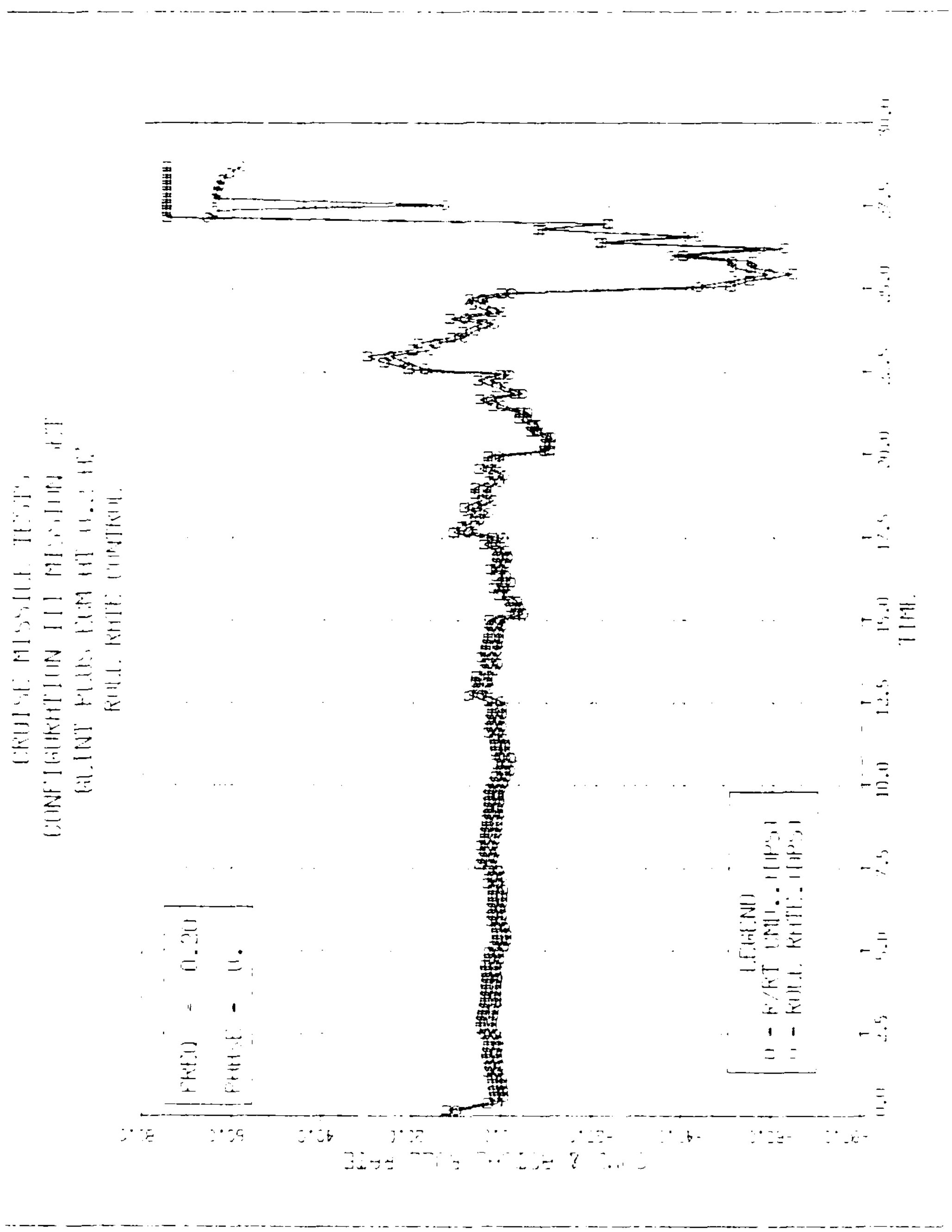


Figure A.66   Conf. III Mission Set - Roll Rate.

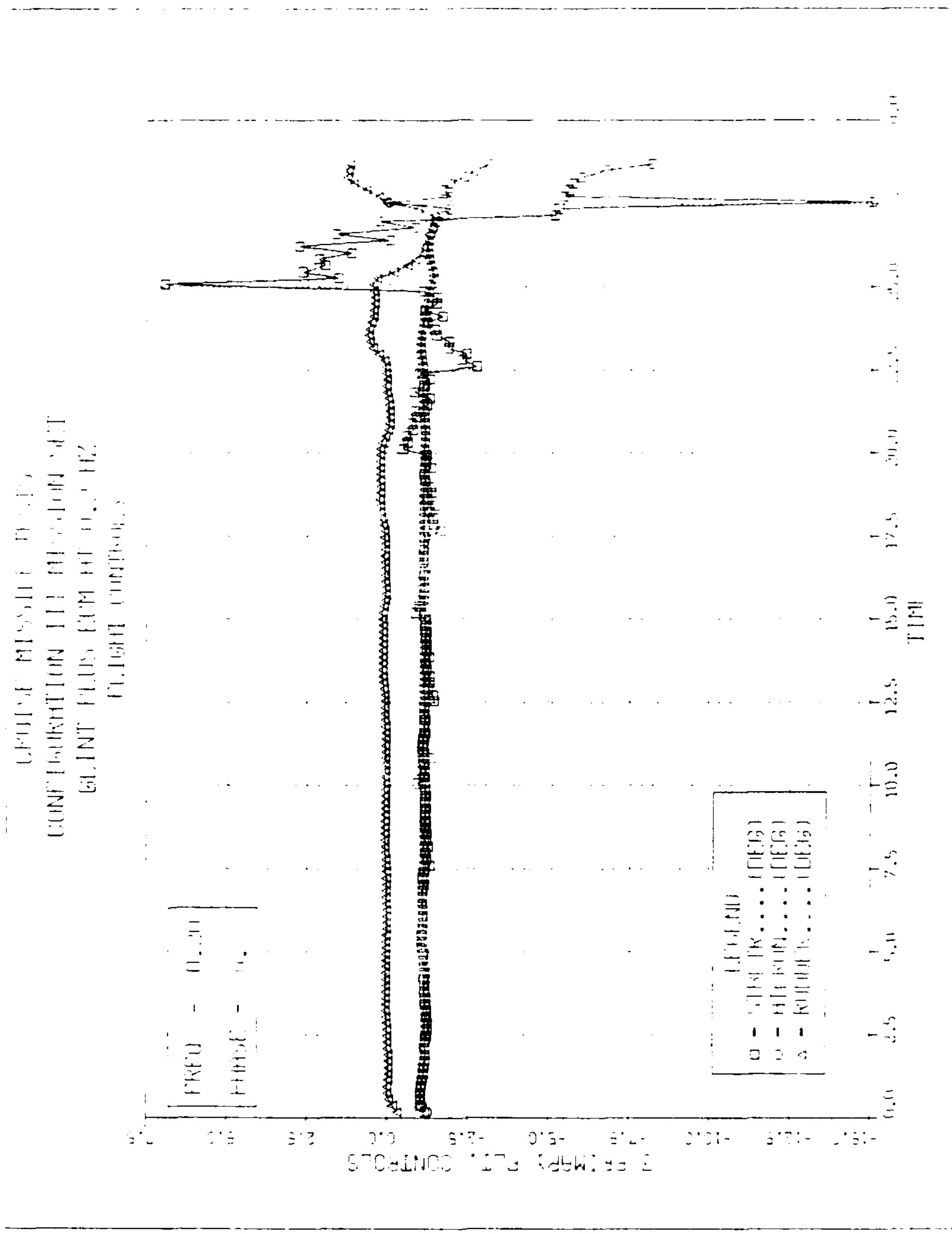


Figure A.67 Conf. III Mission Set - Controls.

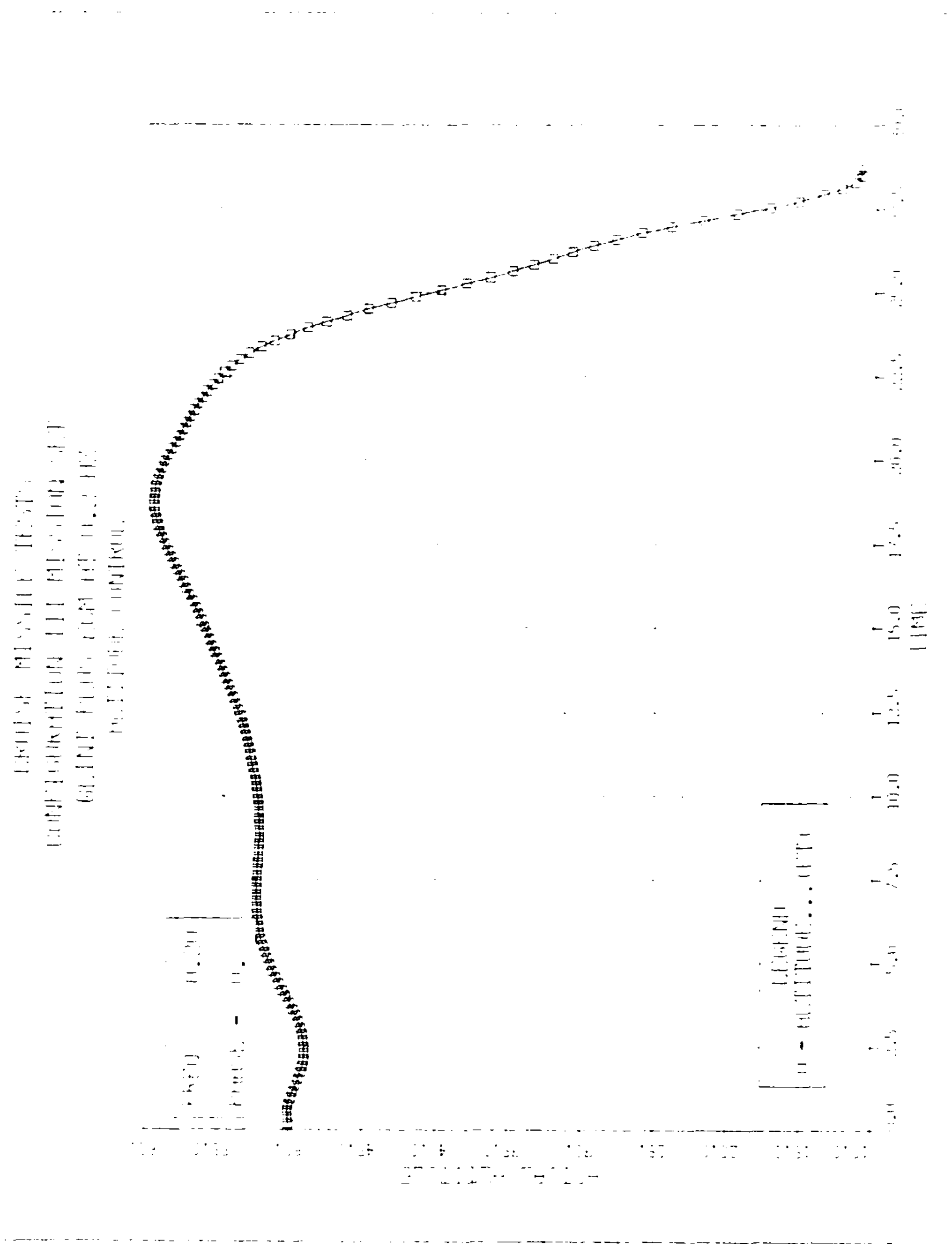


Figure A.68 Conf. III Mission Set - Altitude.

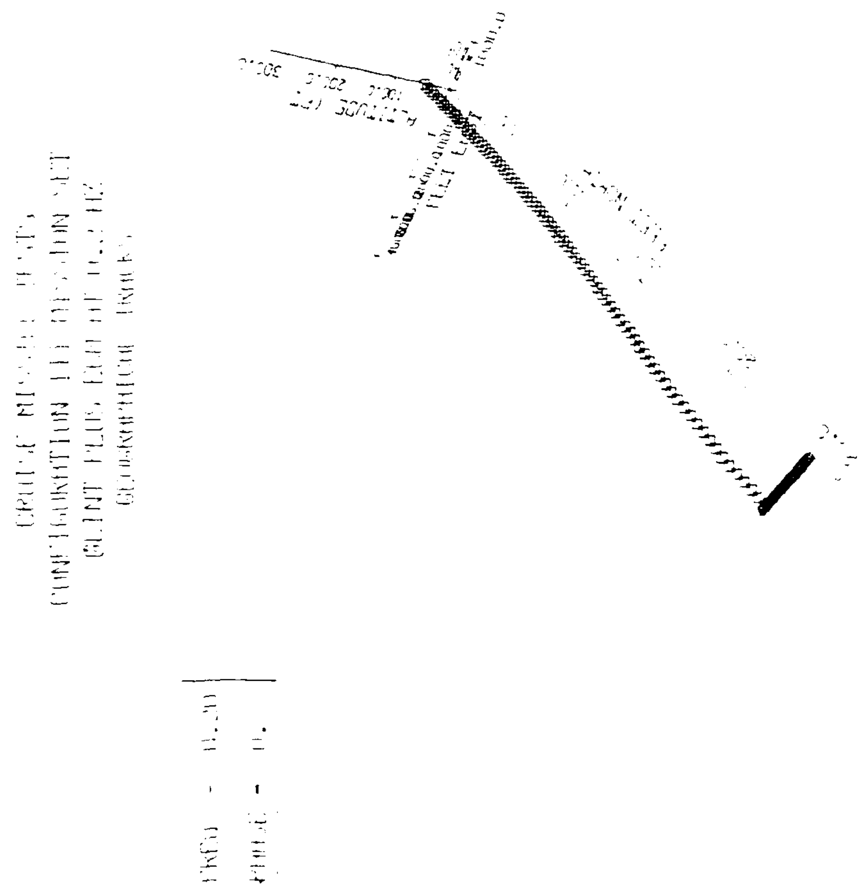


Figure A.69   Conf. III Mission Set - Geo Plot.

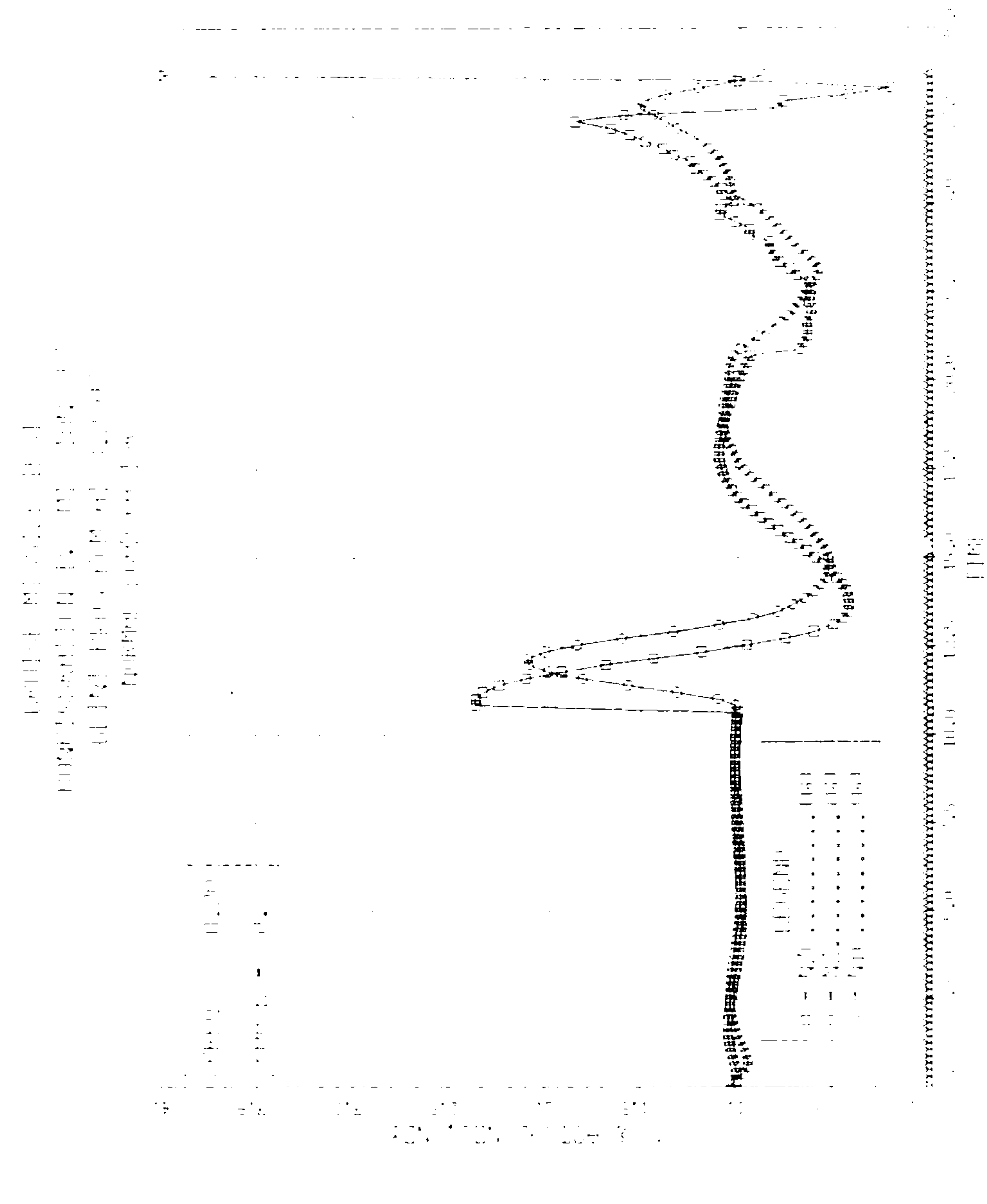


Figure A.70 Conf. IV Mission Set - Load Factor.

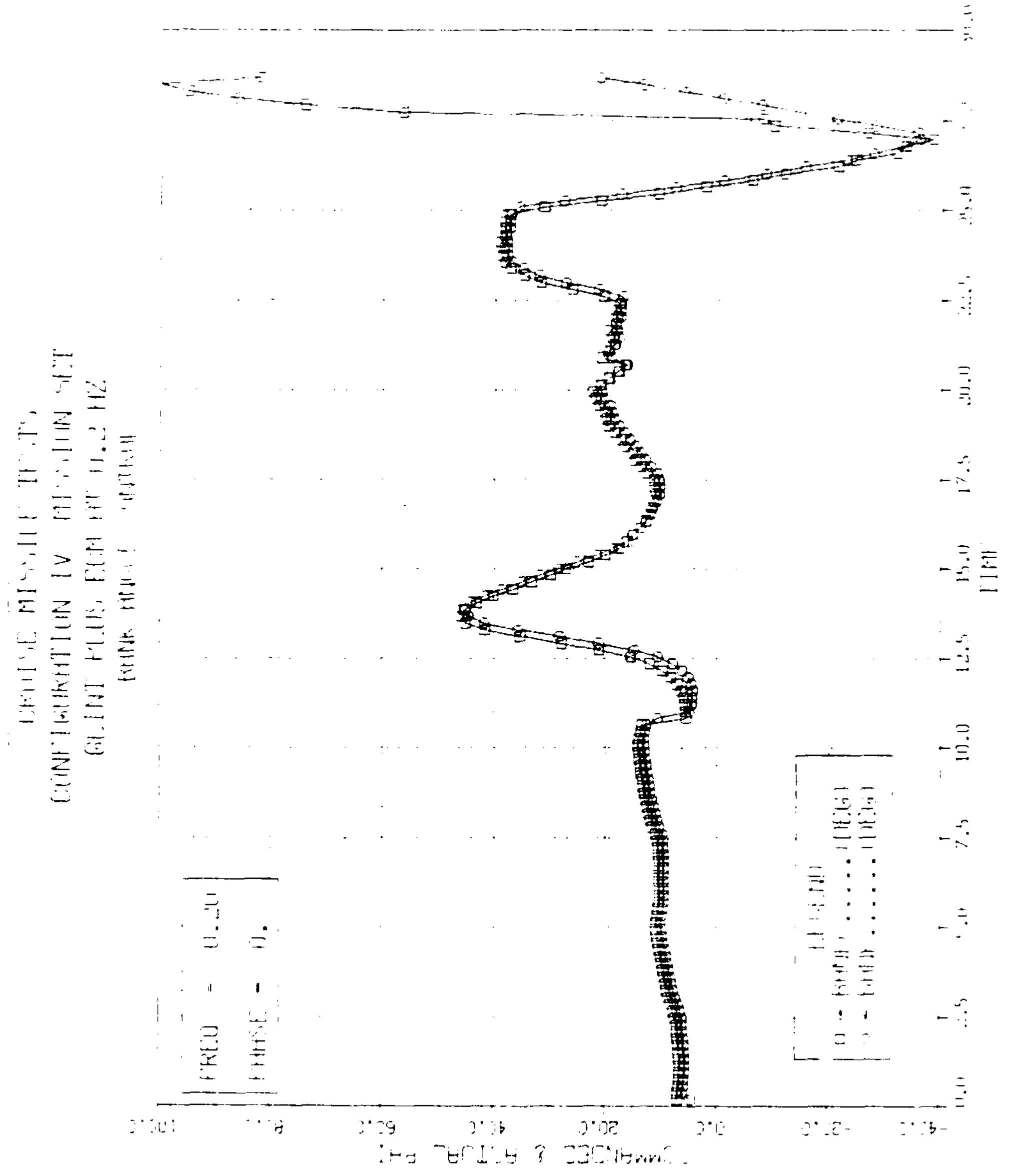


Figure A.71   Conf. IV Mission Set - Bank.

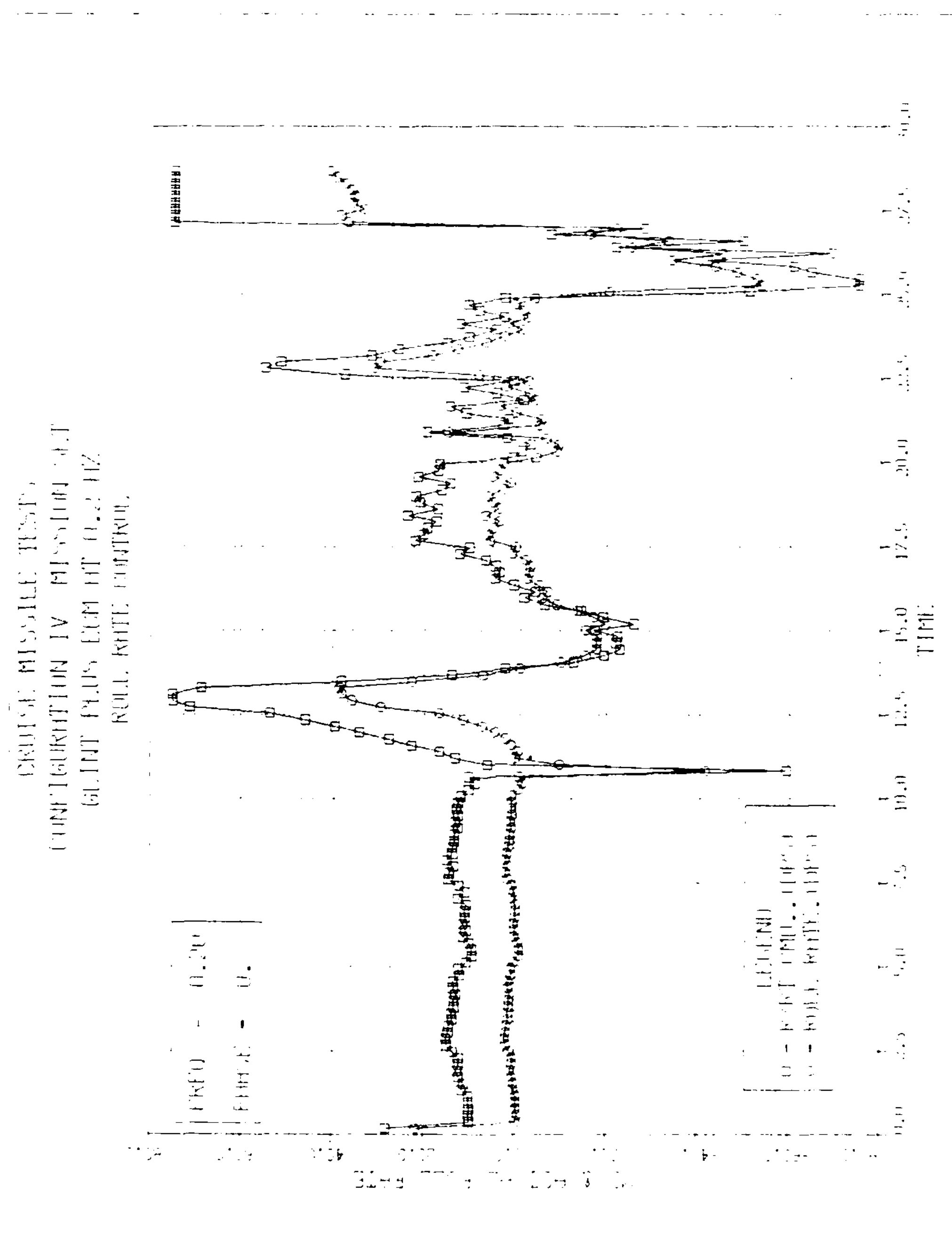


Figure A.72   Conf. IV Mission Set - Roll Rate.

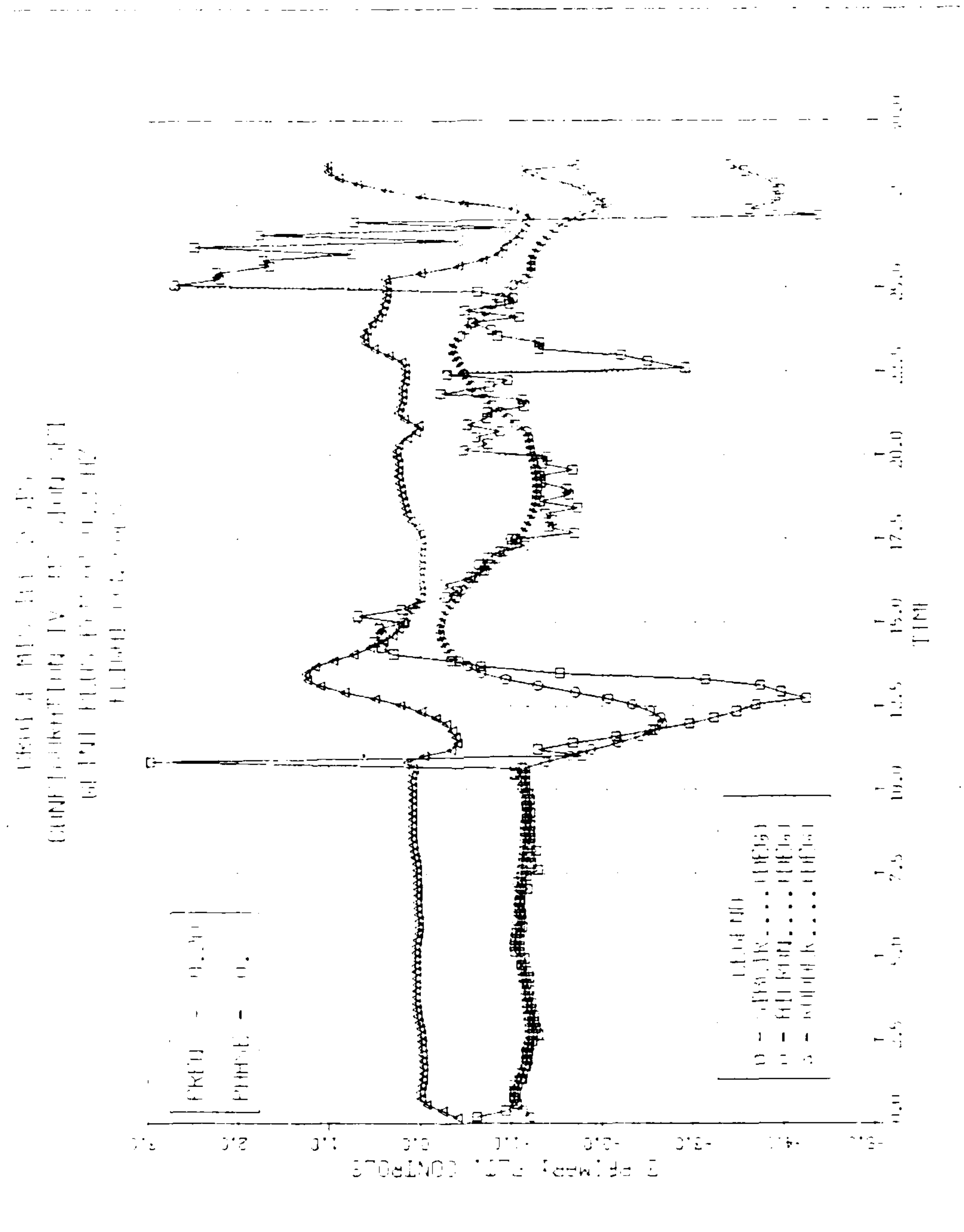


Figure A.73   Conf. IV Mission Set - Controls.

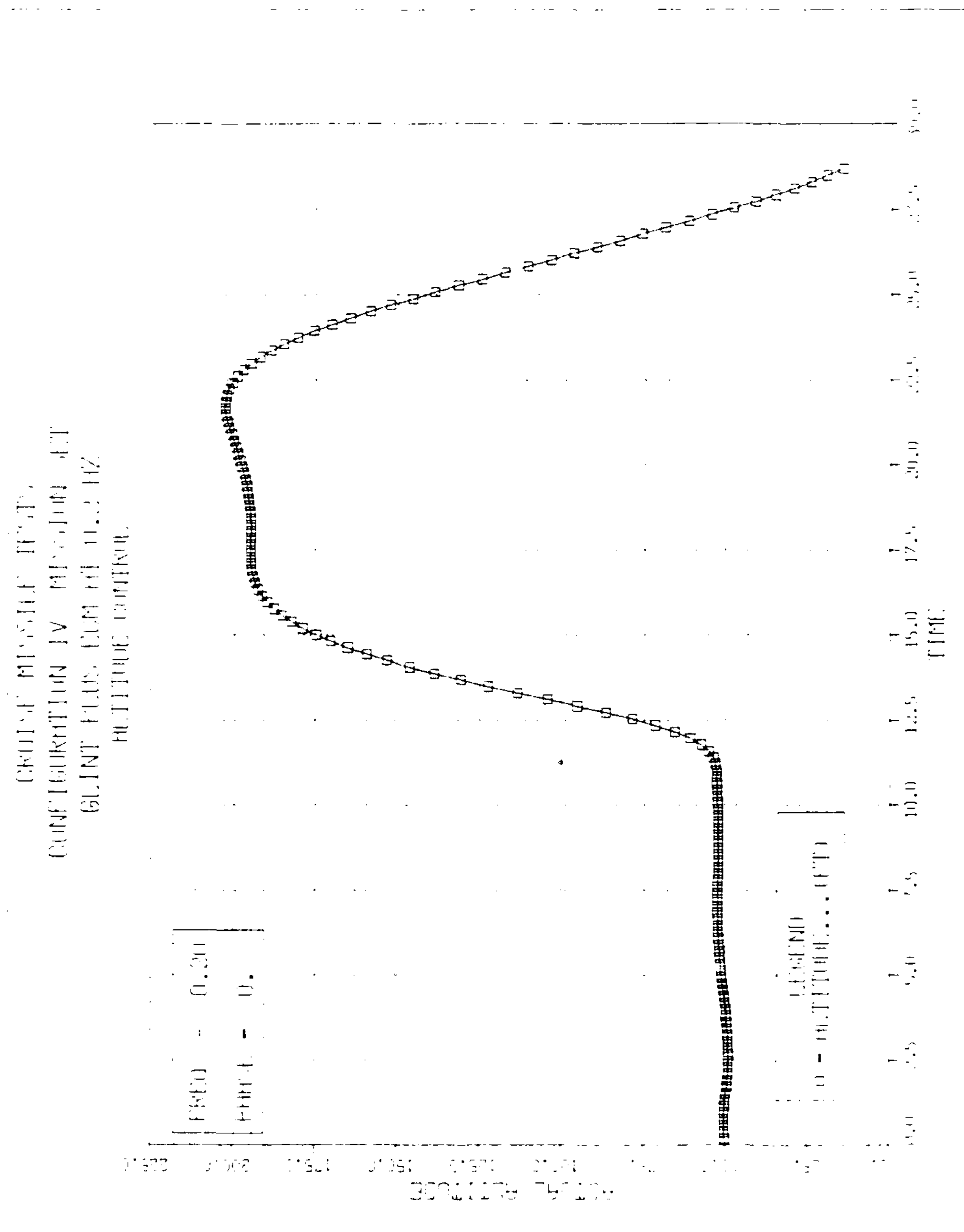


Figure A.74   Conf. IV Mission Set - Altitude.

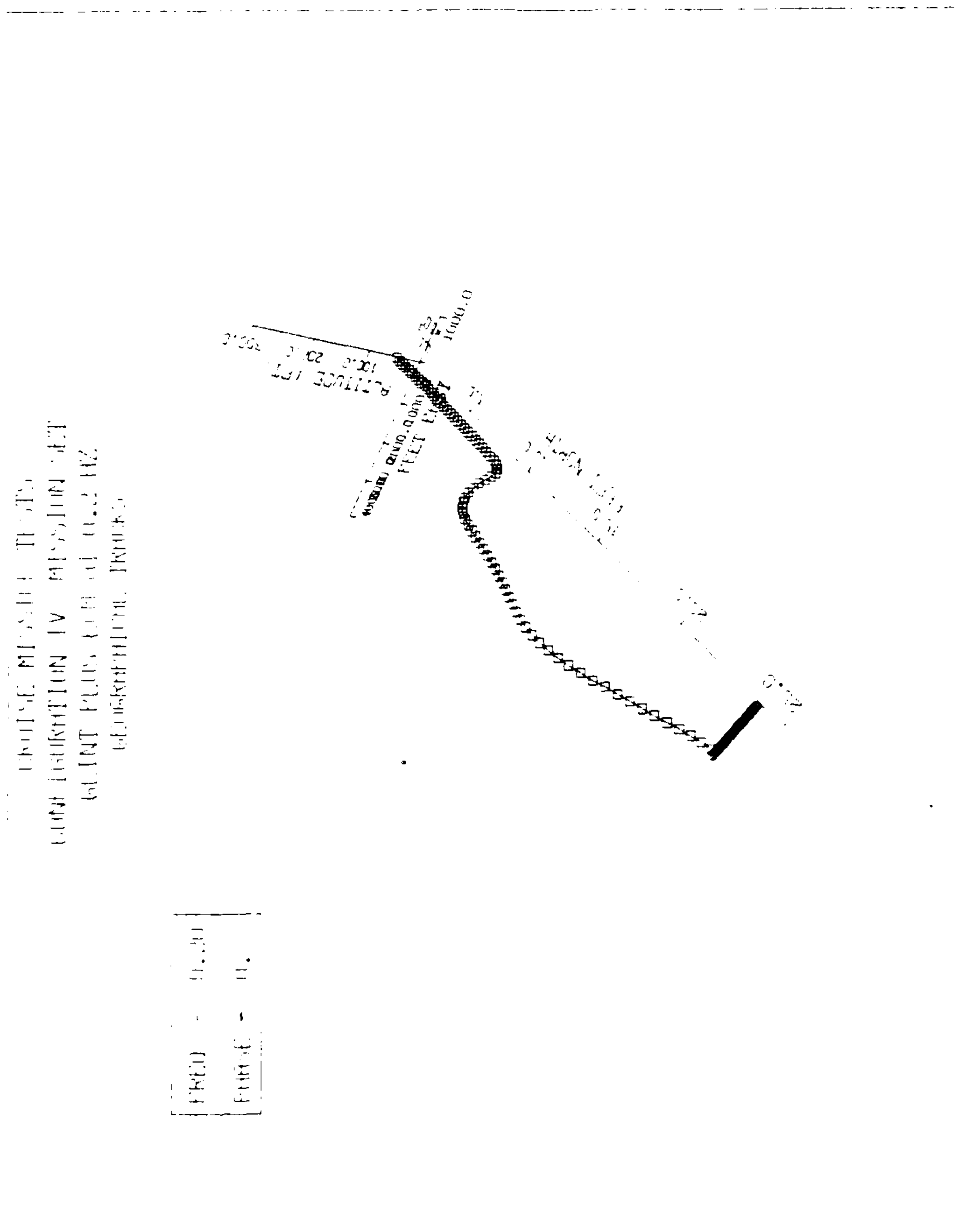


Figure A.75 Conf. IV Mission Set - Geo Plot.

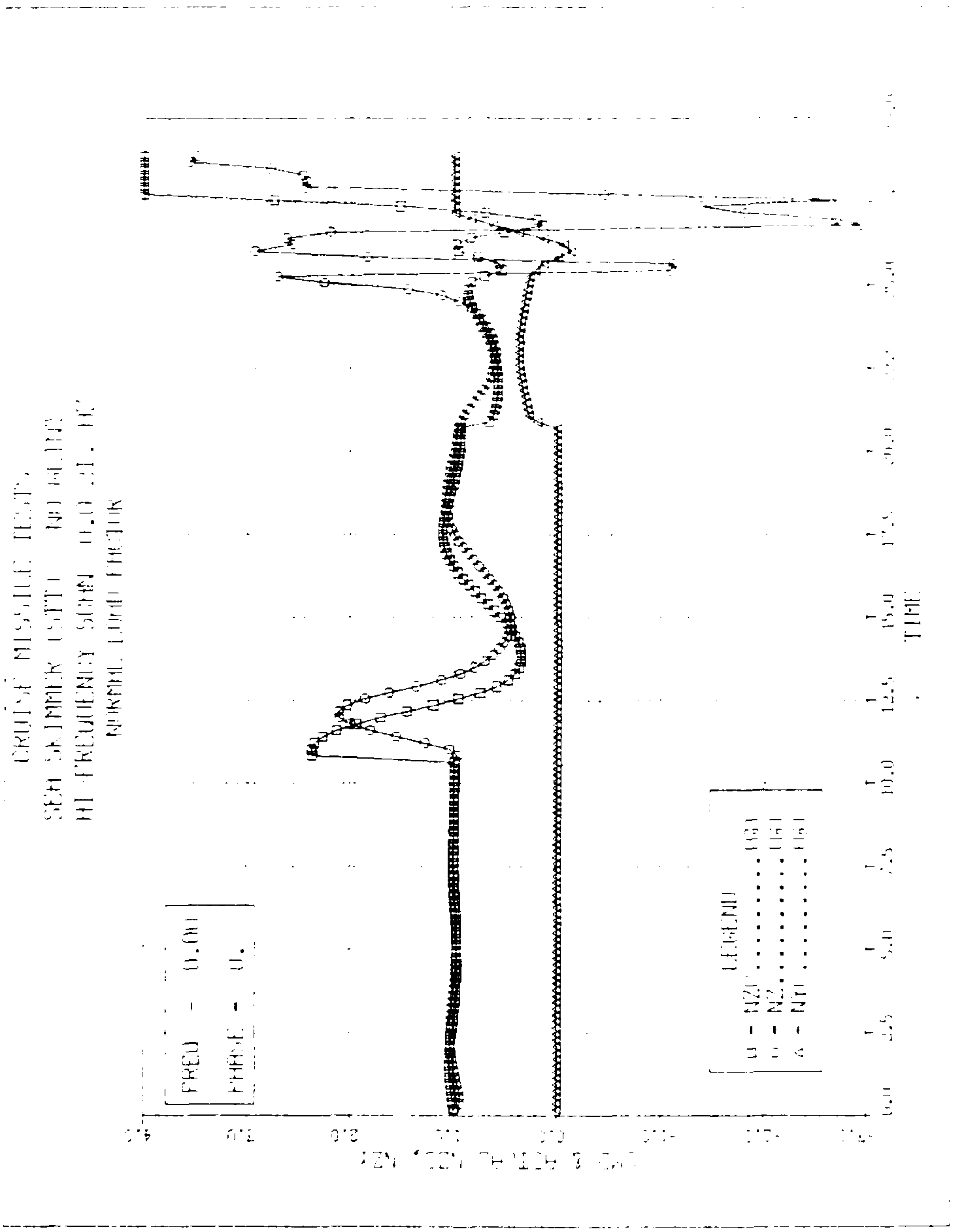


Figure A.76   Conf. V Mission Set - Load Factor.

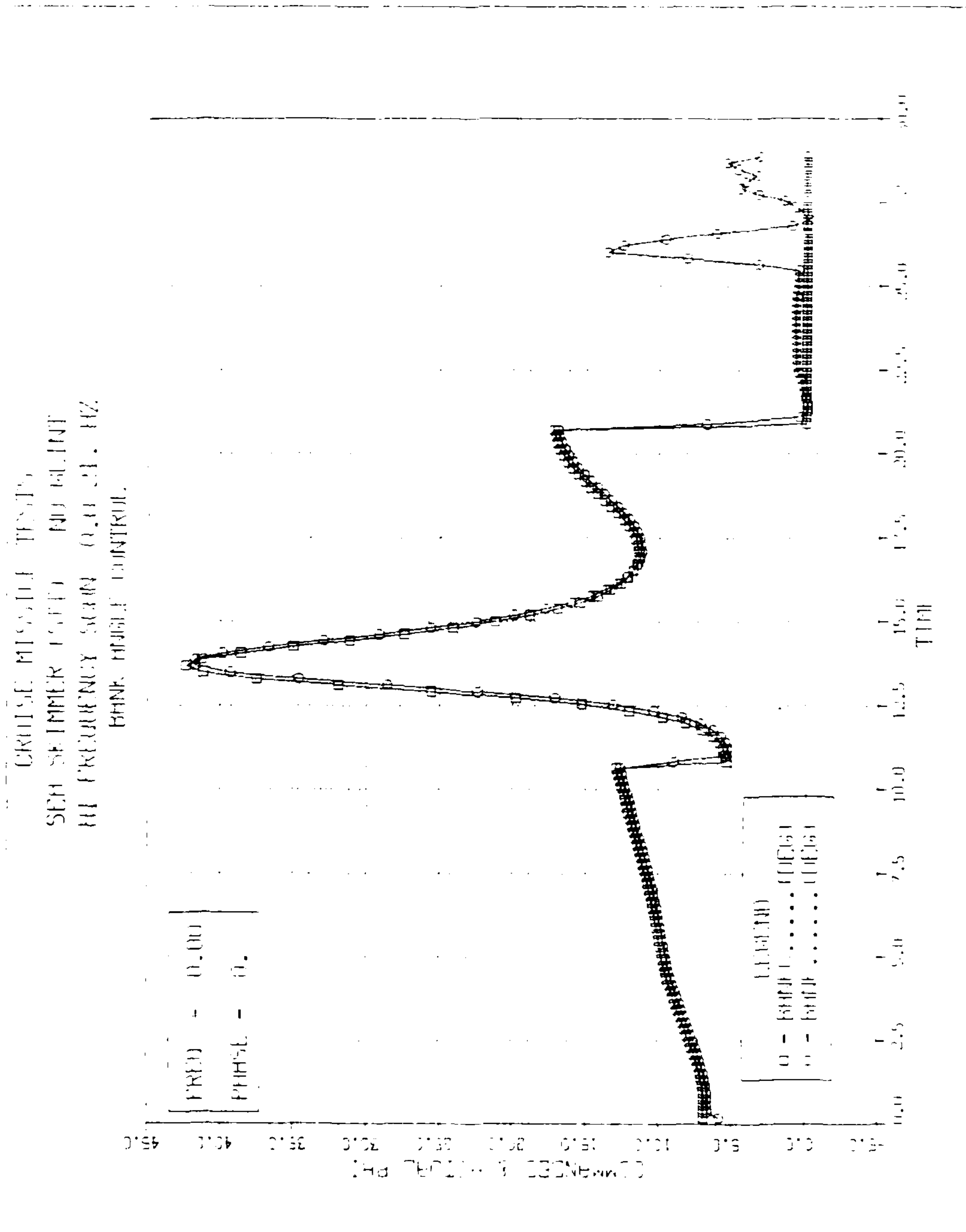


Figure A.77 Conf. V Mission Set - Bank.

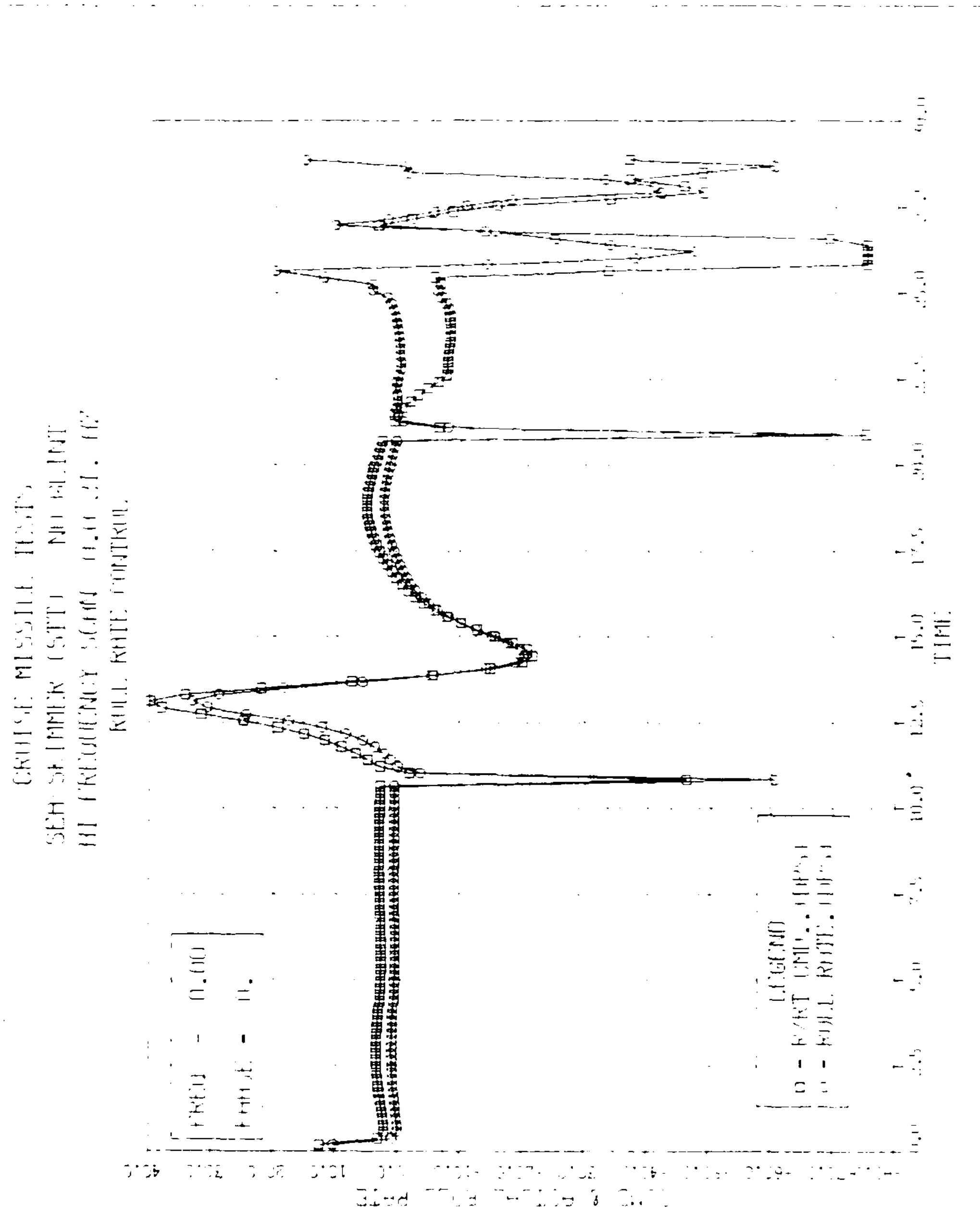


Figure A.78 Conf. V Mission Set - Roll Rate.

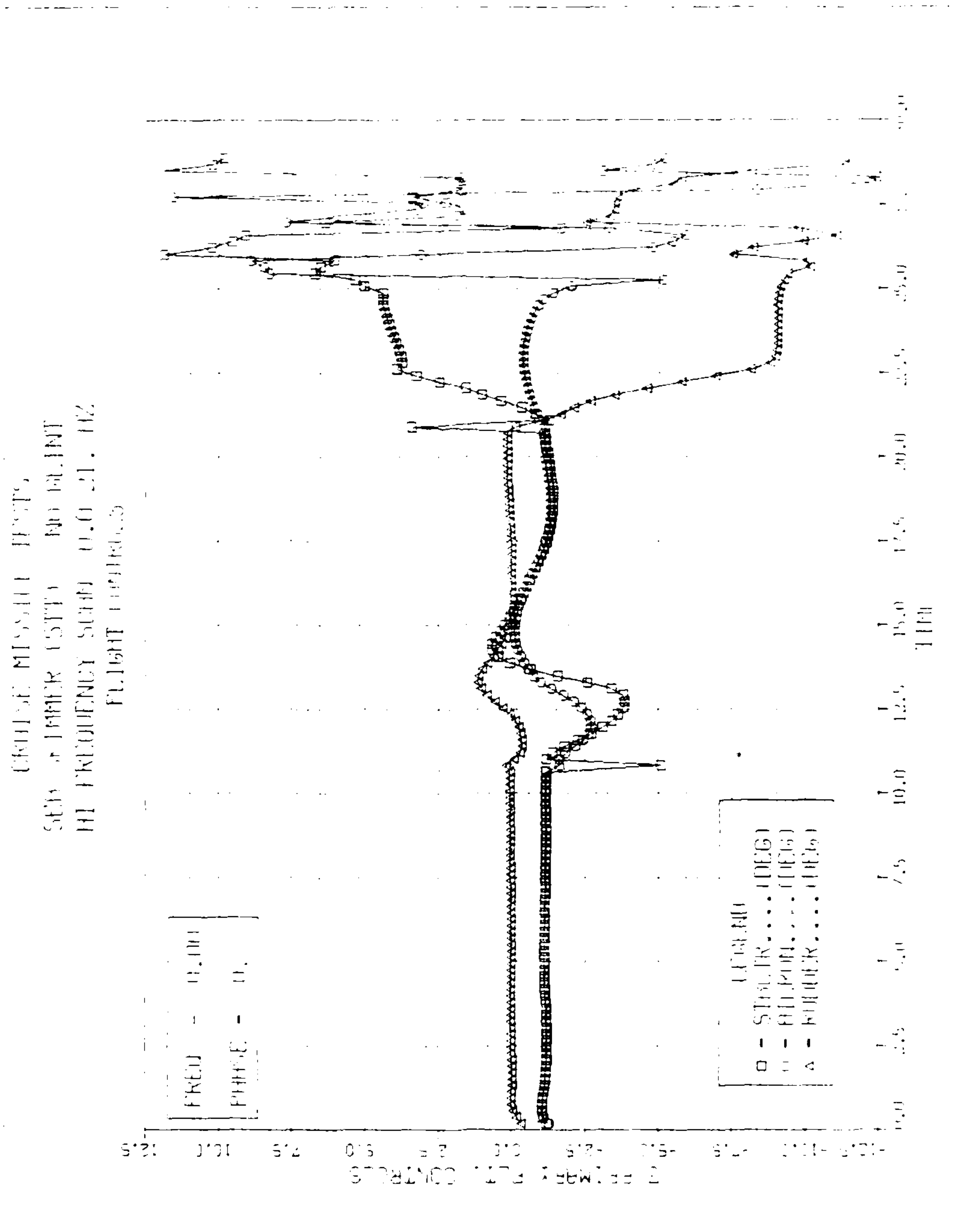


Figure A.79   Conf. V Mission Set - Controls.

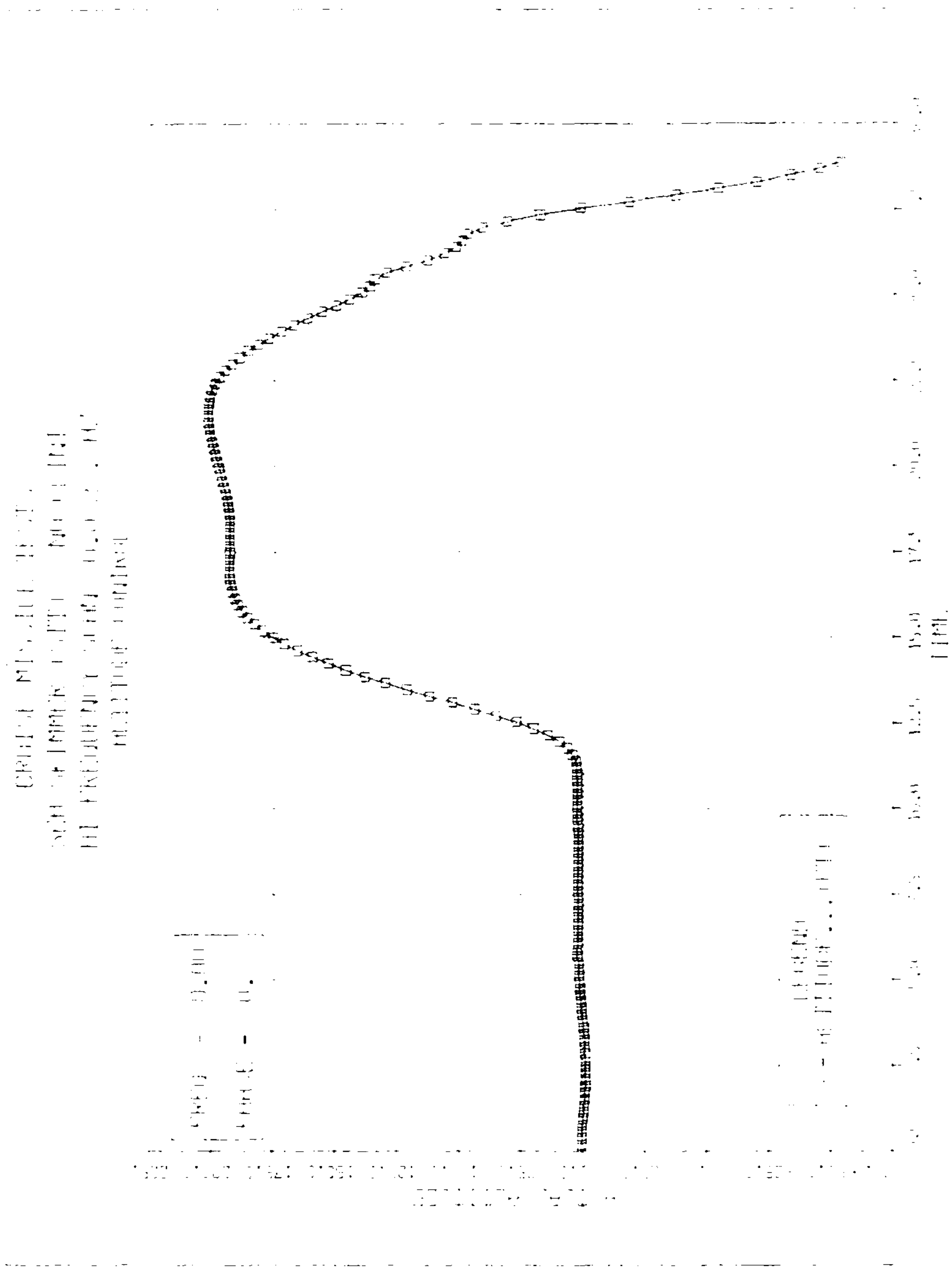


Figure A.80 Conf. V Mission Set - Altitude.

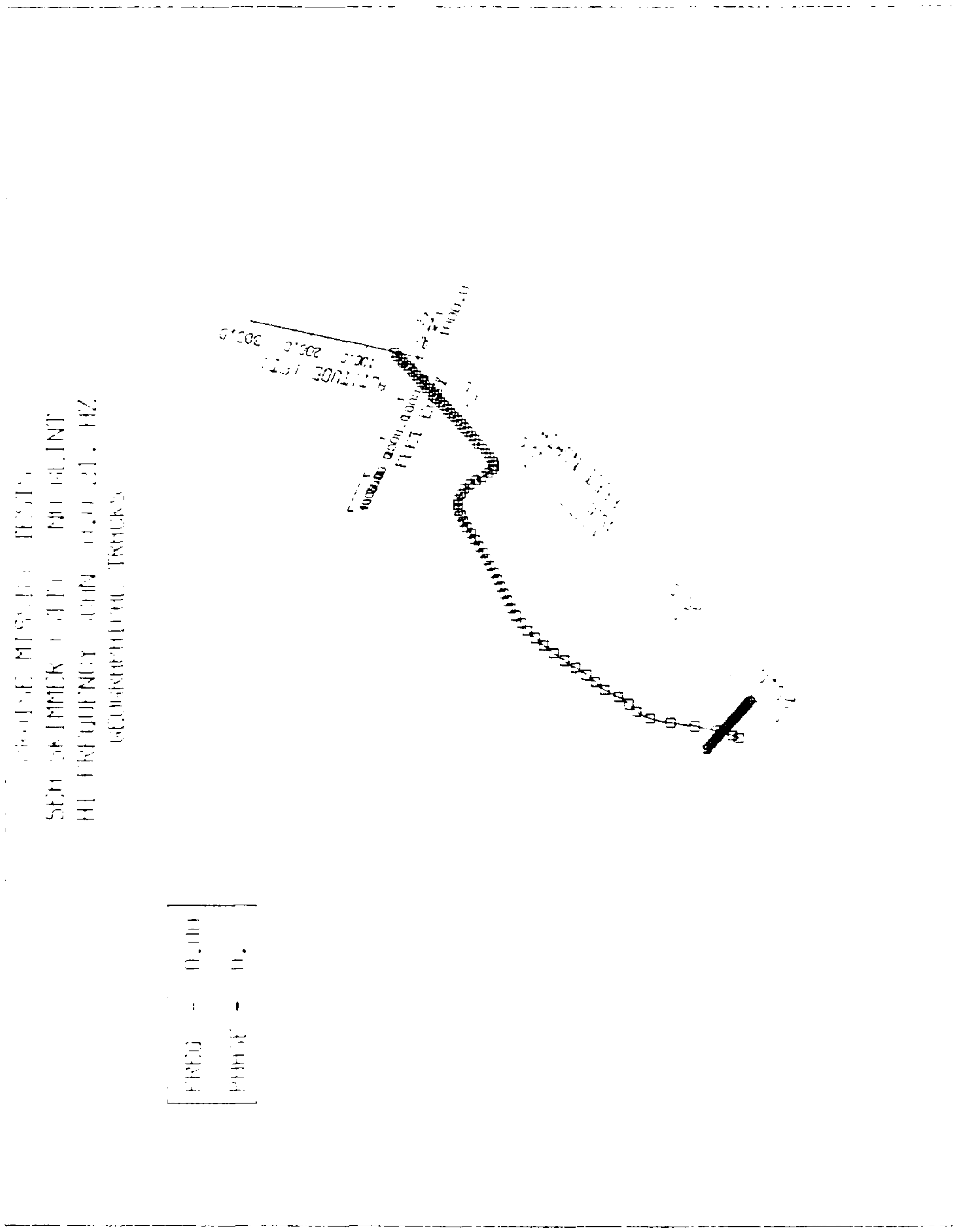


Figure A.81   Conf. V Mission Set - Geo Plot.

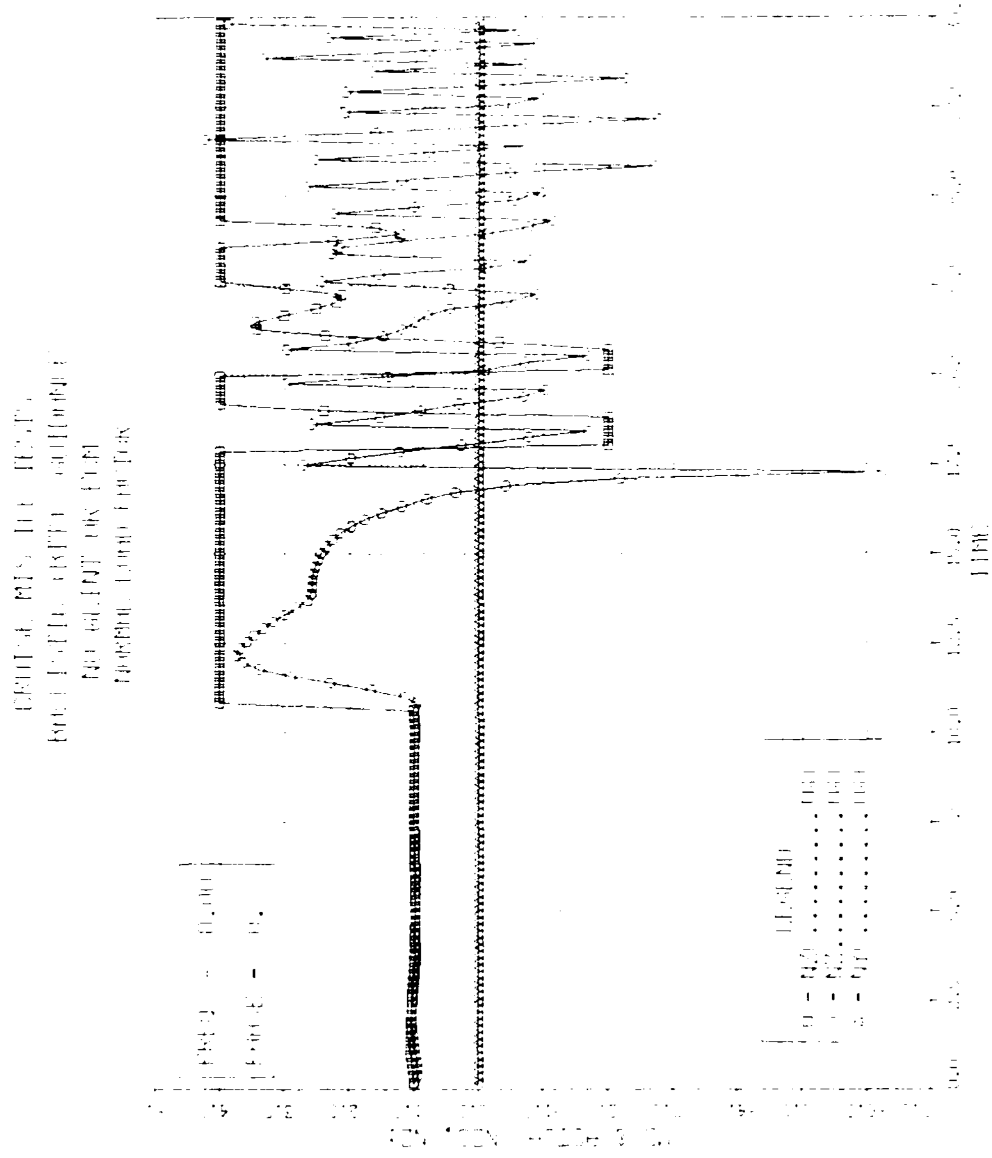


Figure A.82 Conf. VI Mission Set - Load Factor.

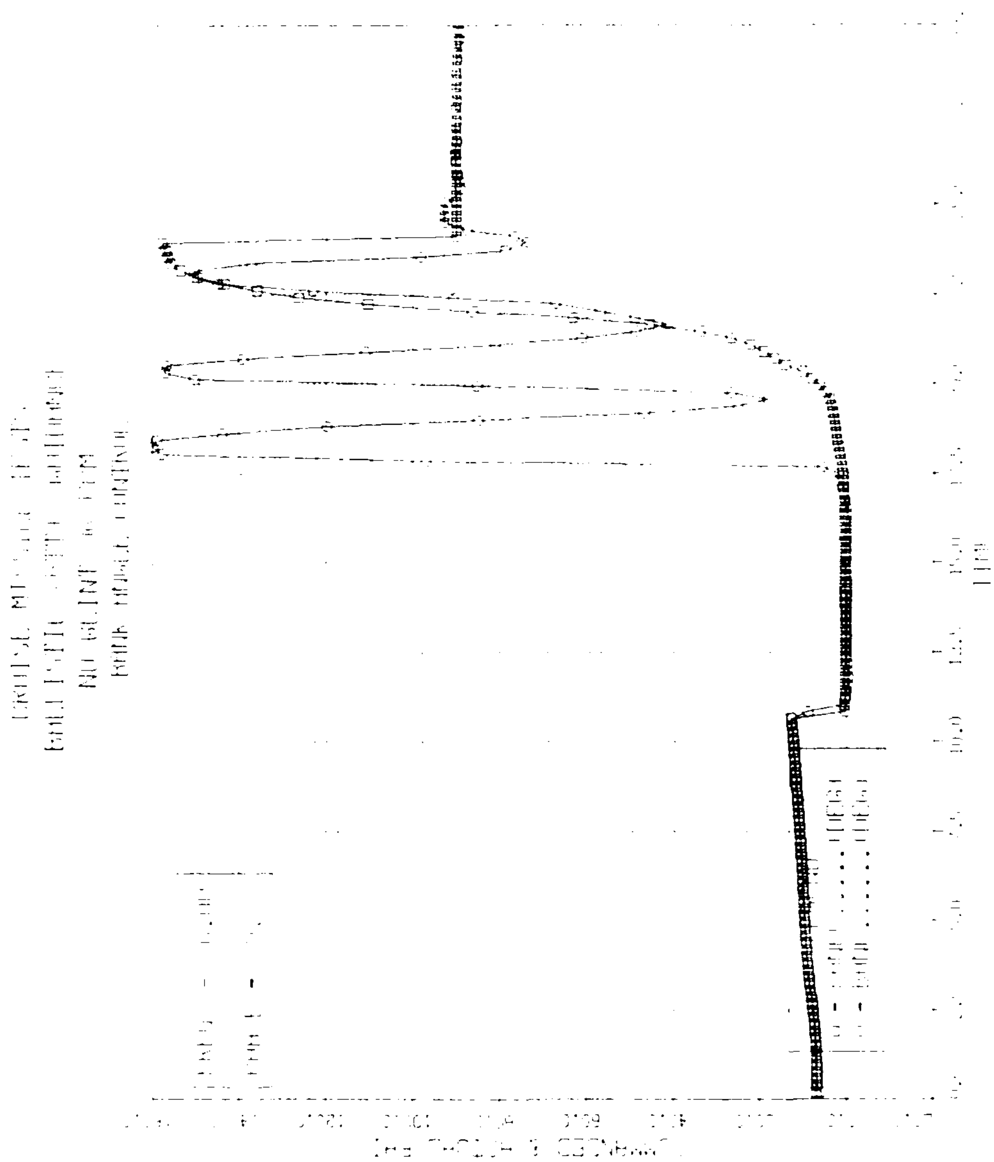


Figure A.83   Conf. VI Mission Set - Bank.

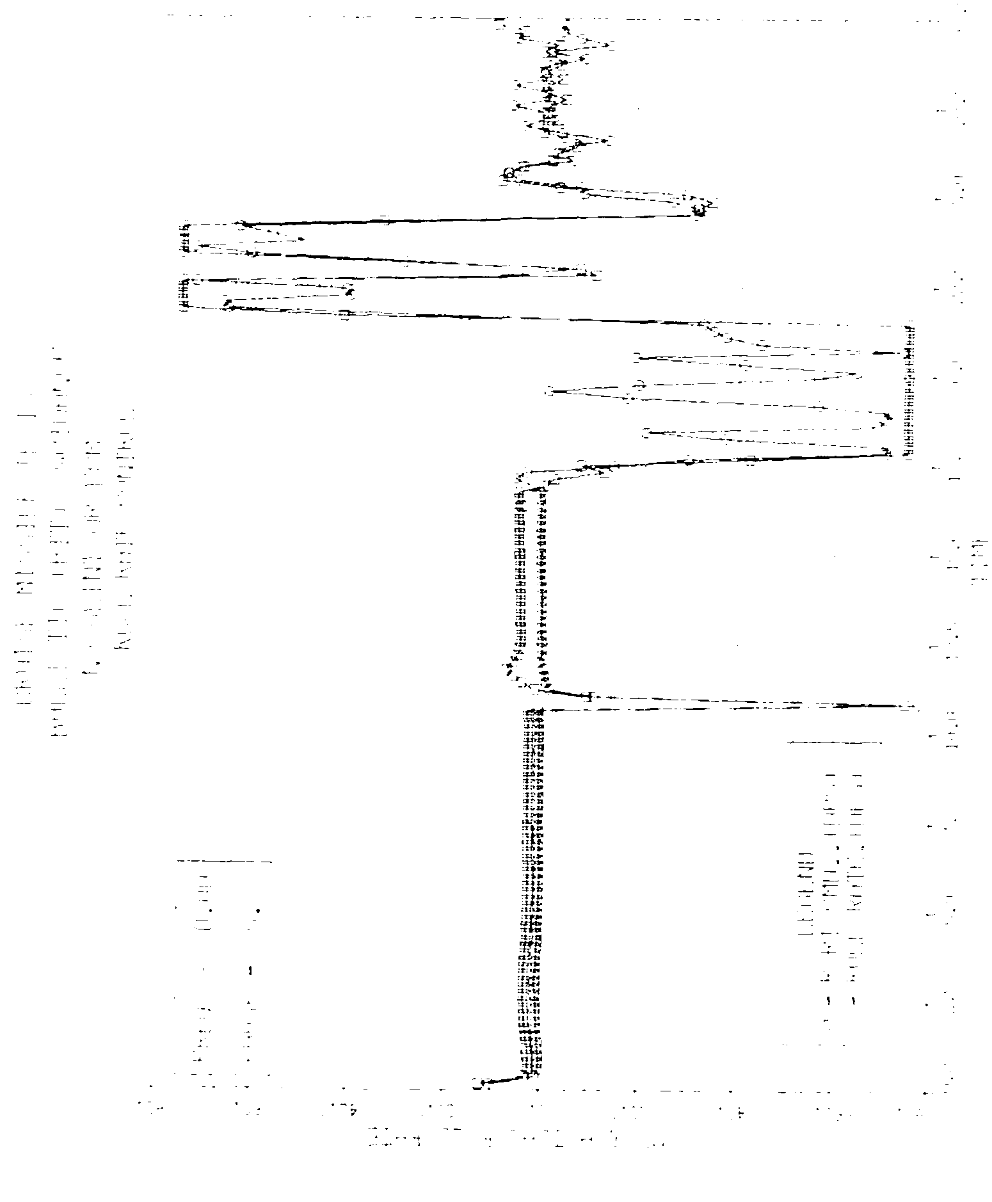


Figure A.84   Conf. VI Mission Set - Roll Rate.

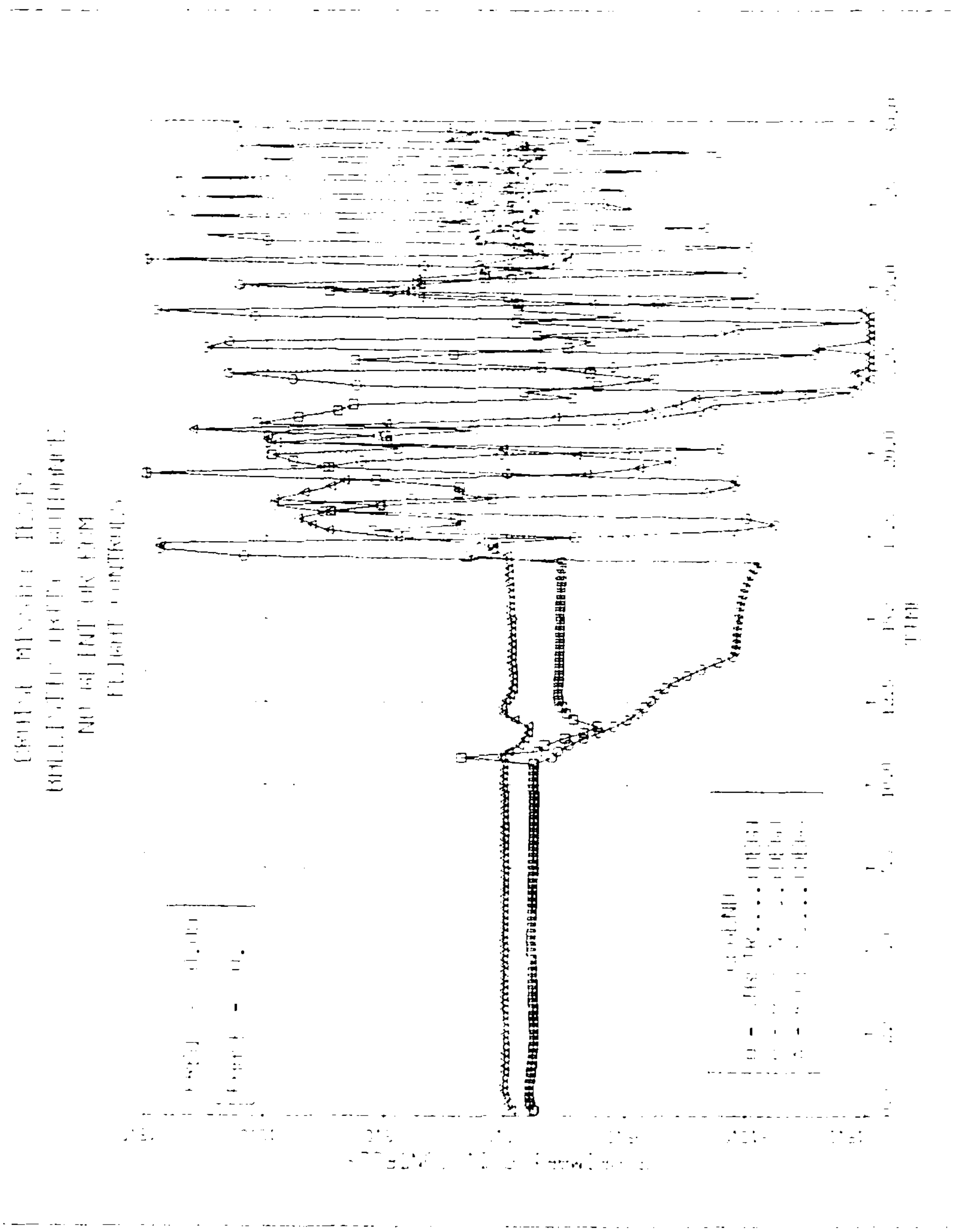


Figure A.85   Conf. VI Mission Set - Controls.

APPENDIX B

SIMULATION PROGRAM TABULAR DATA OUTPUT

```
*****************************************************************************
```

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

```
SIMULATION TERMINATED DUE TO CPA
*** BLINKER FRECUENCY=      0.20
*** BLINKER PHASE    =      0.
```

MISS DISTANCE	*	BANK	*	ROLLRATE	*	AZIMUTH	*	ELEVATION	*
42.87372	*	0.20466	*	0.11202	*	0.03046	*	0.00866	*

(ERROR FUNCTIONS: BANK, ROLLRATE, AZIMUTH, ELEVATION)

******* RANGES FOR ALL SAVED VARIABLES *******

	MINIMUM	MAXIMUM
TIME......(SEC)	0.190000	29.061737
NZC.........(G)	0.289882	4.000000
NZ..........(G)	0.551692	3.750864
BANKC.....(DEG)	-115.634293	92.840149
BANK......(DEG)	-115.765106	84.187943
R/RT CMC..(DPS)	-74.999985	74.999985
ROLL RATE.(DPS)	-73.526184	73.167145
ECM SHIFT..(FT)	-75.000000	75.000000
GLINT SHIFT(FT)	-47.924805	47.070313
STBLTR....(DEG)	-15.000000	7.934396
AILRLN....(DEG)	-6.261881	0.000000

RUDDER....(DEG)	-3.446685	3.952714
ALTITUDE...(FT)	13.361130	250.106537
XM...(FT NORTH)	159.440155	23887.324200
YM....(FT EAST)	-3.526452	1771.405030
XT...(FT NORTH)	24000.000000	24000.000000
XM....(FT EAST)	6.649995	1017.160640
RANGE......(FT)	116.888184	23840.589800
PHASE MARKER	0.000000	4.000000

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 1 OF 4

	TIME......(SEC)		NZC.........(G)		NZ..........(G)		BANKC.....(DEG)		BANK......(DEG)
*	0.190000	*	0.996454	*	0.984984	*	6.838765	*	5.825313
*	0.380000	*	1.002067	*	0.939250	*	6.808969	*	6.542444
*	0.570000	*	1.012286	*	0.935853	*	6.645293	*	6.510144
*	0.759999	*	1.022474	*	0.946949	*	6.546911	*	6.396442
*	0.949999	*	1.030748	*	0.962120	*	6.503017	*	6.327559
*	1.139989	*	1.036386	*	0.977738	*	6.458877	*	6.279079
*	1.329975	*	1.039639	*	0.991862	*	6.589023	*	6.328140
*	1.519961	*	1.040093	*	1.004217	*	6.606047	*	6.386812
*	1.709948	*	1.038448	*	1.014323	*	6.744779	*	6.470633
*	1.899934	*	1.034911	*	1.021769	*	6.918423	*	6.623720
*	2.089920	*	1.029593	*	1.027047	*	7.007813	*	6.747380
*	2.279906	*	1.023017	*	1.030146	*	7.091106	*	6.834517
*	2.469893	*	1.015957	*	1.030586	*	7.323308	*	7.001084
*	2.659879	*	1.008910	*	1.028598	*	7.690206	*	7.254130
*	2.849865	*	1.002075	*	1.024892	*	8.092773	*	7.688922
*	3.039851	*	0.995503	*	1.020577	*	8.427306	*	8.050633
*	3.229837	*	0.989452	*	1.015759	*	8.742263	*	8.377364
*	3.419824	*	0.983882	*	1.010625	*	8.969087	*	8.645678
*	3.609810	*	0.979212	*	1.005136	*	9.224430	*	8.892548
*	3.799796	*	0.975287	*	0.999566	*	9.405898	*	9.108006
*	3.989782	*	0.972403	*	0.994111	*	9.605278	*	9.302847
*	4.179769	*	0.970392	*	0.988990	*	9.752930	*	9.474633
*	4.369755	*	0.969642	*	0.984181	*	9.988882	*	9.674054

*	4.559741	*	0.969938	*	0.979871	*	10.235960	*	9.915122
*	4.749727	*	0.970555	*	0.976926	*	10.297152	*	10.058768
*	4.939713	*	0.972024	*	0.974900	*	10.423241	*	10.161740
*	5.129700	*	0.973534	*	0.973648	*	10.384534	*	10.226965
*	5.319686	*	0.974787	*	0.973969	*	10.169158	*	10.063795
*	5.509672	*	0.976223	*	0.974415	*	9.974829	*	9.865085
*	5.699658	*	0.977961	*	0.974676	*	9.844348	*	9.706358
*	5.889645	*	0.980385	*	0.974635	*	9.898959	*	9.676970
*	6.079631	*	0.982658	*	0.975111	*	9.851148	*	9.668716
*	6.269617	*	0.985226	*	0.976146	*	9.896841	*	9.674454
*	6.459603	*	0.987471	*	0.977697	*	9.856885	*	9.669850
*	6.649590	*	0.989611	*	0.979646	*	9.846107	*	9.647703
*	6.839576	*	0.991370	*	0.981773	*	9.794549	*	9.613314
*	7.029562	*	0.993606	*	0.983401	*	9.991375	*	9.700686
*	7.219548	*	0.998174	*	0.949411	*	59.999954	*	11.834122
*	7.409534	*	1.097366	*	0.834173	*	59.999954	*	24.587097
*	7.599521	*	1.299035	*	0.803401	*	59.999954	*	37.007568
*	7.789507	*	1.677993	*	0.811473	*	59.999954	*	49.419754
*	7.979493	*	2.247486	*	0.916378	*	59.999954	*	58.482132
*	8.169479	*	2.473261	*	1.204178	*	59.999954	*	59.601837
*	8.359466	*	2.579926	*	1.523088	*	59.999954	*	59.680084
*	8.549452	*	2.618698	*	1.830282	*	59.999954	*	59.636536
*	8.739438	*	2.589437	*	2.115276	*	59.999954	*	59.560257
*	8.929424	*	2.495302	*	2.347334	*	59.999954	*	59.475677
*	9.119411	*	2.347077	*	2.504561	*	59.999954	*	59.397644
*	9.309397	*	2.161033	*	2.576699	*	59.999954	*	59.342728
*	9.499383	*	1.955719	*	2.563868	*	59.999954	*	59.316010
*	9.689369	*	1.749104	*	2.477899	*	59.999954	*	59.314651
*	9.879355	*	1.557662	*	2.332685	*	59.999954	*	59.346558
*	10.069342	*	1.394344	*	2.146254	*	59.999954	*	59.401413
*	10.259328	*	1.268426	*	1.938939	*	59.999954	*	59.469284
*	10.449314	*	1.184954	*	1.730132	*	59.999954	*	59.540421
*	10.639300	*	1.144872	*	1.536979	*	59.999954	*	59.606598
*	10.829287	*	0.850376	*	1.371496	*	0.000000	*	47.870846
*	11.019273	*	0.654307	*	1.266437	*	0.000000	*	34.769897
*	11.209259	*	0.533505	*	1.138158	*	0.000000	*	21.793655
*	11.399245	*	0.464801	*	1.010703	*	0.000000	*	8.865244
*	11.589231	*	0.440546	*	0.917152	*	0.000000	*	0.877188
*	11.779218	*	0.437432	*	0.851973	*	0.000000	*	-0.043441

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 1 OF 4

	TIME......(SEC)		NZC.........(G)		NZ..........(G)		BANKC.....(DEG)		BANK......(DEG)
*	11.969204	*	0.450394	*	0.768242	*	0.000000	*	-0.161884
*	12.159190	*	0.482046	*	0.685690	*	0.000000	*	-0.162543
*	12.349176	*	0.531986	*	0.614517	*	0.000000	*	-0.139814
*	12.539163	*	0.597679	*	0.569422	*	0.000000	*	-0.120249
*	12.729149	*	0.674333	*	0.551692	*	0.000000	*	-0.108467
*	12.919135	*	0.757205	*	0.557850	*	0.000000	*	-0.104236
*	13.109121	*	0.841999	*	0.584029	*	0.000000	*	-0.105790
*	13.299108	*	0.924948	*	0.626355	*	0.000000	*	-0.111387
*	13.489094	*	1.002849	*	0.679794	*	0.000000	*	-0.119782
*	13.679080	*	1.073686	*	0.738611	*	0.000000	*	-0.132744
*	13.869066	*	1.135511	*	0.804878	*	0.000000	*	-0.148205
*	14.059052	*	1.186506	*	0.874575	*	0.000000	*	-0.164867
*	14.249039	*	1.225576	*	0.943938	*	0.000000	*	-0.181834
*	14.439025	*	1.252302	*	1.009762	*	0.000000	*	-0.198240
*	14.629011	*	1.266858	*	1.069501	*	0.000000	*	-0.213442
*	14.818997	*	1.269832	*	1.123638	*	0.000000	*	-0.226920
*	15.008984	*	1.261657	*	1.169641	*	0.000000	*	-0.238204
*	15.198970	*	1.243847	*	1.203627	*	0.000000	*	-0.246988
*	15.388956	*	1.218340	*	1.225422	*	0.000000	*	-0.253153
*	15.578942	*	1.187195	*	1.235477	*	0.000000	*	-0.256717
*	15.768929	*	1.152465	*	1.234756	*	0.000000	*	-0.257825
*	15.958915	*	1.116103	*	1.224596	*	0.000000	*	-0.256717
*	16.148834	*	1.079892	*	1.206601	*	0.000000	*	-0.253706
*	16.338730	*	1.045380	*	1.182526	*	0.000000	*	-0.249153
*	16.528625	*	1.013837	*	1.154176	*	0.000000	*	-0.243445
*	16.718521	*	0.986244	*	1.123310	*	0.000000	*	-0.236972
*	16.908417	*	0.963268	*	1.092164	*	0.000000	*	-0.230110
*	17.098312	*	0.944965	*	1.063701	*	0.000000	*	-0.223170
*	17.288208	*	0.931297	*	1.036421	*	0.000000	*	-0.216391
*	17.478104	*	0.922266	*	1.011223	*	0.000000	*	-0.210006
*	17.667999	*	0.917679	*	0.988825	*	0.000000	*	-0.204210
*	17.857895	*	0.917206	*	0.969775	*	0.000000	*	-0.199162
*	18.047791	*	0.920379	*	0.954433	*	0.000000	*	-0.194973
*	18.237686	*	0.926643	*	0.942978	*	0.000000	*	-0.191712
*	18.427582	*	0.935383	*	0.935418	*	0.000000	*	-0.189403
*	18.617477	*	0.945946	*	0.931596	*	0.000000	*	-0.188026
*	18.807373	*	0.957685	*	0.931231	*	0.000000	*	-0.187525
*	18.997269	*	0.969978	*	0.933928	*	0.000000	*	-0.187627
*	19.187164	*	0.982247	*	0.939216	*	0.000000	*	-0.188814
*	19.377060	*	4.000000	*	0.956647	*	-19.212936	*	-11.716519

*	19.566956	*	4.000000	*	1.336203	*	-19.645432	*	-18.829987
*	19.756851	*	4.000000	*	1.953647	*	-20.514221	*	-20.287033
*	19.946747	*	4.000000	*	2.603067	*	-21.803818	*	-21.606384
*	20.136642	*	4.000000	*	3.165137	*	-23.917358	*	-23.416321
*	20.326538	*	4.000000	*	3.547853	*	-26.628387	*	-25.876419
*	20.516434	*	3.993346	*	3.707689	*	-29.875137	*	-28.782227
*	20.706329	*	3.645613	*	3.750864	*	-33.380920	*	-32.118164
*	20.896225	*	3.328323	*	3.695785	*	-37.201050	*	-35.788803
*	21.086121	*	1.394300	*	3.743388	*	-86.270721	*	-41.193802
*	21.276016	*	1.548610	*	3.527078	*	-91.988892	*	-54.749466
*	21.465912	*	1.641810	*	3.080334	*	-97.266113	*	-67.457657
*	21.655807	*	1.692579	*	2.546545	*	-102.053802	*	-80.066727
*	21.845703	*	1.737501	*	2.077121	*	-106.587250	*	-92.754532
*	22.035595	*	1.724108	*	1.733549	*	-109.825348	*	-105.360977
*	22.225494	*	1.644519	*	1.687409	*	-111.873642	*	-110.729401
*	22.415390	*	1.576145	*	1.724775	*	-113.156738	*	-112.752167
*	22.605286	*	1.476245	*	1.736968	*	-114.502014	*	-113.975922
*	22.795181	*	1.340478	*	1.709265	*	-115.634293	*	-115.439880
*	22.985077	*	1.156681	*	1.647637	*	-115.251144	*	-115.765106
*	23.174973	*	1.009784	*	1.550969	*	-114.067169	*	-115.013672
*	23.364868	*	0.838435	*	1.428517	*	-110.889008	*	-112.811676
*	23.554764	*	0.704791	*	1.297051	*	-106.212891	*	-108.906342
*	23.744659	*	0.574909	*	1.162587	*	-98.385605	*	-102.649292
*	23.934555	*	0.480631	*	1.033371	*	-87.793304	*	-93.434158
*	24.124451	*	0.341435	*	0.913162	*	-61.075912	*	-81.214630

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 1 OF 4

TIME......(SEC)		NZC..........(G)		NZ...........(G)		BANKC.....(DEG)		BANK......(DEG)	
*	24.314346	*	0.312690	*	0.819130	*	-41.207794	*	-68.769150
*	24.504242	*	0.289882	*	0.748993	*	-12.722169	*	-56.286392
*	24.694138	*	0.323168	*	0.693249	*	7.391577	*	-43.769882
*	24.884033	*	0.411515	*	0.642170	*	14.987059	*	-31.229416
*	25.073929	*	0.560545	*	0.600143	*	13.566038	*	-18.679733
*	25.263824	*	0.793154	*	0.574242	*	11.106000	*	-6.130711

*	25.453720	*	0.933357	*	0.581695	*	7.270296	* 5.453405
*	25.643616	*	1.042595	*	0.699352	*	5.325726	* 5.767602
*	25.833511	*	1.131395	*	0.796295	*	3.147260	* 3.985226
*	26.023407	*	1.231064	*	0.873536	*	2.354088	* 2.600244
*	26.213303	*	1.275287	*	0.949175	*	-1.190448	* 0.250407
*	26.403198	*	1.373239	*	1.020734	*	-1.089107	* -1.234868
*	26.593094	*	1.414298	*	1.093149	*	-3.873116	* -2.835041
*	26.782990	*	1.498616	*	1.170446	*	-3.584943	* -3.880977
*	26.972885	*	1.534853	*	1.246133	*	-5.545876	* -4.924376
*	27.162781	*	1.728691	*	1.302967	*	1.781992	* -1.654127
*	27.352676	*	2.026227	*	1.329571	*	10.870007	* 6.405899
*	27.542572	*	1.853992	*	1.533800	*	14.335903	* 8.333620
*	27.732468	*	2.109669	*	1.368543	*	48.266494	* 19.853433
*	27.922363	*	3.284542	*	1.460158	*	64.898895	* 31.915710
*	28.112259	*	4.000000	*	1.654933	*	76.102646	* 43.756592
*	28.302155	*	4.000000	*	2.019216	*	85.941116	* 55.290161
*	28.492050	*	4.000000	*	2.458335	*	92.840149	* 66.378754
*	28.681946	*	4.000000	*	2.864714	*	84.516556	* 77.037064
*	28.871841	*	4.000000	*	3.297837	*	87.914276	* 84.187943
*	29.061737	*	4.000000	*	3.747063	*	78.001495	* 81.152344

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 2 OF 4

	TIME......(SEC)		R/RT CMD..(DPS)		ROLL RATE.(DPS)		ECM SHIFT..(FT)	GLINT SHIFT(FT)
*	0.190000	*	12.067198	*	9.865781	*	-75.000000	* 19.921875
*	0.380000	*	2.970557	*	0.744763	*	-75.000000	* 47.070313
*	0.570000	*	1.389124	*	-0.660753	*	-75.000000	* -43.310547
*	0.759995	*	1.571068	*	-0.488012	*	-75.000000	* -25.952148
*	0.949995	*	1.868235	*	-0.235456	*	-75.000000	* -19.880780
*	1.139985	*	1.916773	*	-0.222514	*	-75.000000	* -47.924805
*	1.329975	*	2.881227	*	0.608558	*	-75.000000	* 45.671082
*	1.519961	*	2.381867	*	0.131129	*	-75.000000	* -43.750000
*	1.709948	*	3.031082	*	0.685417	*	-75.000000	* 19.897461
*	1.899934	*	3.278272	*	0.890394	*	-75.000000	* 32.194409
*	2.089920	*	2.867801	*	0.508729	*	-75.000000	* -27.539063

*	2.279906	*	2.820111	*	0.457421	*	-75.000000	*	-42.953164
*	2.469893	*	3.609996	*	1.165082	*	-75.000000	*	37.622070
*	2.659879	*	4.559071	*	2.016505	*	75.000000	*	-5.271912
*	2.849865	*	4.583967	*	2.048492	*	75.000000	*	2.807617
*	3.039851	*	4.257809	*	1.761084	*	75.000000	*	-9.204102
*	3.229837	*	4.120385	*	1.648042	*	75.000000	*	-0.720215
*	3.419824	*	3.626225	*	1.217953	*	75.000000	*	-33.416748
*	3.609810	*	3.728097	*	1.322030	*	75.000000	*	-5.682373
*	3.799796	*	3.323970	*	0.973187	*	75.000000	*	-33.227539
*	3.989782	*	3.378778	*	1.033268	*	75.000000	*	-14.042664
*	4.179769	*	3.092050	*	0.787648	*	75.000000	*	-31.372070
*	4.369755	*	3.529077	*	1.189073	*	75.000000	*	22.955322
*	4.559741	*	3.603025	*	1.265257	*	75.000000	*	43.412781
*	4.749727	*	2.618767	*	0.387185	*	75.000000	*	-34.065247
*	4.939713	*	2.893435	*	0.636503	*	75.000000	*	2.099609
*	5.129700	*	1.653059	*	-0.484199	*	-75.000000	*	31.295776
*	5.319686	*	1.026196	*	-1.037718	*	-75.000000	*	-12.695313
*	5.509672	*	1.077647	*	-0.988593	*	-75.000000	*	-28.857422
*	5.699658	*	1.415164	*	-0.684597	*	-75.000000	*	-24.087524
*	5.889645	*	2.419119	*	0.215088	*	-75.000000	*	42.173767
*	6.079631	*	1.948160	*	-0.209821	*	-75.000000	*	-10.717773
*	6.269617	*	2.423877	*	0.211000	*	-75.000000	*	21.826172
*	6.459603	*	2.002267	*	-0.169043	*	-75.000000	*	-22.363281
*	6.649590	*	2.136669	*	-0.053513	*	-75.000000	*	-19.866780
*	6.839576	*	1.931415	*	-0.240801	*	-75.000000	*	-47.924805
*	7.029562	*	3.238107	*	0.927199	*	-75.000000	*	45.671082
*	7.219548	*	74.999985	*	73.167145	*	-75.000000	*	-43.750000
*	7.409534	*	74.999985	*	65.331711	*	-75.000000	*	19.897461
*	7.599521	*	74.999985	*	65.214294	*	75.000000	*	32.394409
*	7.789507	*	74.999985	*	65.097687	*	75.000000	*	-27.539063
*	7.979493	*	18.041362	*	13.599403	*	75.000000	*	-42.953164
*	8.169479	*	4.459596	*	1.007195	*	75.000000	*	37.622070
*	8.359466	*	3.449299	*	-0.345169	*	75.000000	*	-5.271912
*	8.549452	*	3.887762	*	-0.645887	*	75.000000	*	2.807617
*	8.739438	*	4.701549	*	-0.759738	*	75.000000	*	-9.204102
*	8.929424	*	5.614478	*	-0.785816	*	75.000000	*	-0.720215
*	9.119411	*	6.464853	*	-0.724356	*	75.000000	*	-33.416748
*	9.309397	*	7.075636	*	-0.545664	*	75.000000	*	-5.682373
*	9.499383	*	7.377189	*	-0.398665	*	75.000000	*	-33.227539
*	9.689369	*	7.410828	*	-0.180096	*	75.000000	*	-14.042664
*	9.879355	*	7.082125	*	0.013767	*	75.000000	*	-31.372070
*	10.069342	*	6.500261	*	0.155098	*	-75.000000	*	22.955322
*	10.259328	*	5.771453	*	0.234564	*	-75.000000	*	43.412781
*	10.449314	*	5.002514	*	0.256450	*	-75.000000	*	-34.065247
*	10.639300	*	4.281966	*	0.208768	*	-75.000000	*	2.099609
*	10.829287	*	-74.999985	*	-69.740738	*	-75.000000	*	31.295776
*	11.019273	*	-74.999985	*	-68.475403	*	-75.000000	*	-12.695313
*	11.209259	*	-74.999985	*	-68.066742	*	-75.000000	*	-28.857422

*	11.399245	*	-74.999985	*	-67.783478	*	-75.000000	*	-24.087524
*	11.589231	*	-10.854013	*	-11.366899	*	-75.000000	*	42.173767
*	11.779218	*	0.309608	*	-1.175964	*	-75.000000	*	-10.717773

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 2 OF 4

	TIME......(SEC)		R/RT CMD..(DPS)		ROLL RATE.(DPS)		ECM SHIFT..(FT)		GLINT SHIFT(FT)
*	11.969204	*	1.725770	*	-0.077368	*	-75.000000	*	21.826172
*	12.159190	*	1.766047	*	0.153380	*	-75.000000	*	-22.363281
*	12.349176	*	1.522860	*	0.138551	*	-75.000000	*	-19.886780
*	12.539163	*	1.307849	*	0.090493	*	75.000000	*	-47.924805
*	12.729149	*	1.175968	*	0.044148	*	75.000000	*	45.671082
*	12.919135	*	1.126412	*	0.008935	*	75.000000	*	-43.750000
*	13.109121	*	1.140409	*	-0.015377	*	75.000000	*	19.857461
*	13.299108	*	1.198962	*	-0.031489	*	75.000000	*	32.354409
*	13.489094	*	1.287829	*	-0.049077	*	75.000000	*	-27.539063
*	13.679080	*	1.425424	*	-0.068103	*	75.000000	*	-42.953164
*	13.869066	*	1.591445	*	-0.075865	*	75.000000	*	37.622070
*	14.059052	*	1.771146	*	-0.078154	*	75.000000	*	-5.271912
*	14.249039	*	1.954254	*	-0.075742	*	75.000000	*	2.807617
*	14.439025	*	2.131940	*	-0.069467	*	75.000000	*	-9.204102
*	14.629011	*	2.296958	*	-0.060213	*	75.000000	*	-0.720215
*	14.818997	*	2.443621	*	-0.048663	*	75.000000	*	-33.416748
*	15.008984	*	2.566842	*	-0.035191	*	-75.000000	*	-5.682373
*	15.198970	*	2.663178	*	-0.021065	*	-75.000000	*	-33.227539
*	15.388956	*	2.731251	*	-0.007072	*	-75.000000	*	-14.042664
*	15.578942	*	2.771195	*	0.006111	*	-75.000000	*	-31.372070
*	15.768929	*	2.784501	*	0.017965	*	-75.000000	*	22.955322
*	15.958915	*	2.773720	*	0.028139	*	-75.000000	*	43.412781
*	16.148834	*	2.742187	*	0.036394	*	-75.000000	*	-34.065247
*	16.338730	*	2.693789	*	0.042629	*	-75.000000	*	2.099609
*	16.528625	*	2.632695	*	0.046833	*	-75.000000	*	31.295776
*	16.718521	*	2.563117	*	0.049088	*	-75.000000	*	-12.655313
*	16.908417	*	2.489133	*	0.049557	*	-75.000000	*	-28.857422
*	17.098312	*	2.414161	*	0.048846	*	-75.000000	*	-24.087524

17.288208	2.340792	0.046959	-75.000000	42.173767
17.478104	2.271552	0.044037	-75.000000	-10.717773
17.667999	2.208578	0.040282	75.000000	21.826172
17.857895	2.153597	0.035907	75.000000	-22.363281
18.047791	2.107845	0.031138	75.000000	-19.886780
18.237686	2.072089	0.026206	75.000000	-47.924805
18.427582	2.046605	0.021321	75.000000	45.671082
18.617477	2.031221	0.016687	75.000000	-43.750000
18.807373	2.025372	0.012457	75.000000	19.897461
18.997269	2.028170	0.008771	75.000000	32.394409
19.187164	2.038470	0.005710	75.000000	-27.539063
19.377060	-74.999985	-68.719757	75.000000	-42.993164
19.566956	-10.174731	-11.349103	75.000000	37.622070
19.756851	-3.140329	-5.873646	75.000000	-5.271912
19.946747	-2.969819	-7.032695	75.000000	2.807617
20.136642	-6.683754	-10.891152	-75.000000	-9.204102
20.326538	-9.487894	-11.762302	-75.000000	-0.720215
20.516434	-13.608355	-15.634974	-75.000000	-33.416748
20.706329	-15.620285	-17.100998	-75.000000	-5.682373
20.896225	-17.409729	-18.505905	-75.000000	-33.227539
21.086121	-74.999985	-73.526184	-75.000000	-14.042664
21.276016	-74.999985	-68.097229	-75.000000	-31.372070
21.465912	-74.999985	-64.827148	-75.000000	22.955322
21.655807	-74.999985	-65.806488	-75.000000	43.412781
21.845703	-74.999985	-66.102875	-75.000000	-34.065247
22.035599	-54.749130	-55.254349	-75.000000	2.099609
22.225494	-13.985446	-14.129050	-75.000000	31.295776
22.415390	-5.177193	-7.009793	-75.000000	-12.695313
22.605286	-6.651922	-8.923970	75.000000	-28.857422
22.795181	-2.722851	-5.446668	75.000000	-24.087524
22.985077	5.737814	2.087615	75.000000	42.173767
23.174973	10.913829	6.743303	75.000000	-10.717773
23.364868	22.559418	17.168839	75.000000	21.826172
23.554764	31.783112	25.545227	75.000000	-22.363281
23.744659	50.506699	42.293365	75.000000	-19.886780
23.934555	66.983032	57.138718	75.000000	-47.924805
24.124451	74.999985	65.371765	75.000000	45.671082

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 2 OF 4

	TIME......(SEC)		R/RT CMD..(DPS)		ROLL RATE.(DPS)		ECM SHIFT..(FT)		GLINT SHIFT(FT)
*	24.314346	*	74.999985	*	65.562851	*	75.000000	*	-43.750000
*	24.504242	*	74.999985	*	65.786819	*	75.000000	*	19.897401
*	24.694138	*	74.999985	*	65.964462	*	75.000000	*	32.394405
*	24.884033	*	74.999985	*	66.059204	*	75.000000	*	-27.539063
*	25.073929	*	74.999985	*	66.085938	*	-75.000000	*	-42.953164
*	25.263824	*	74.999985	*	66.059158	*	-75.000000	*	37.622070
*	25.453720	*	22.935425	*	22.956116	*	-75.000000	*	-5.271912
*	25.643616	*	-5.669250	*	-6.440586	*	-75.000000	*	2.807617
*	25.833511	*	-10.200819	*	-10.561381	*	-75.000000	*	-9.204102
*	26.023407	*	-3.224416	*	-4.461498	*	-75.000000	*	-0.720215
*	26.213303	*	-17.424454	*	-17.331482	*	-75.000000	*	-33.416748
*	26.403198	*	1.460428	*	-0.733393	*	-75.000000	*	-5.682373
*	26.593094	*	-12.643291	*	-13.444024	*	-75.000000	*	-33.227539
*	26.782990	*	3.221104	*	0.505559	*	-75.000000	*	-14.042664
*	26.972885	*	-7.714266	*	-9.377487	*	-75.000000	*	-31.372070
*	27.162781	*	40.710358	*	33.784058	*	-75.000000	*	22.955322
*	27.352676	*	53.080215	*	44.799210	*	-75.000000	*	45.412781
*	27.542572	*	66.124893	*	25.035751	*	75.000000	*	-34.065247
*	27.732468	*	74.999985	*	63.833176	*	75.000000	*	2.099609
*	27.922363	*	74.999985	*	62.675156	*	75.000000	*	31.295776
*	28.112259	*	74.999985	*	61.318130	*	75.000000	*	-12.655313
*	28.302155	*	74.999985	*	58.976959	*	75.000000	*	-28.857422
*	28.492050	*	74.999985	*	56.138641	*	75.000000	*	-24.087524
*	28.681946	*	74.999985	*	54.246460	*	0.000000	*	42.173767
*	28.871841	*	43.046890	*	24.959106	*	0.000000	*	-10.717773
*	29.061737	*	-38.403168	*	-41.171707	*	0.000000	*	21.826172

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FRECUENCY= 0.20

DATA SET NUMBER 3 OF 4

TIME......(SEC) STBLTR....(DEG) AILRON....(DEG) RUDDER....(DEG) ALTITUDE...(FT)

*	0.190000	*	-1.179925	*	-1.272073	*	-0.320945	*	50.047150
*	0.380000	*	-1.038112	*	-1.095920	*	-0.182107	*	50.089844
*	0.570000	*	-1.017799	*	-1.048470	*	-0.056974	*	50.075516
*	0.759999	*	-1.034143	*	-1.054128	*	-0.009282	*	49.959283
*	0.949999	*	-1.055637	*	-1.077138	*	-0.005256	*	49.873306
*	1.139989	*	-1.071321	*	-1.104088	*	-0.014276	*	49.714828
*	1.329975	*	-1.142786	*	-1.129882	*	-0.022971	*	49.541748
*	1.519961	*	-1.122277	*	-1.152071	*	-0.024183	*	49.370300
*	1.709948	*	-1.177706	*	-1.170511	*	-0.022619	*	49.214798
*	1.899934	*	-1.195317	*	-1.184159	*	-0.018078	*	49.086792
*	2.089920	*	-1.177351	*	-1.193243	*	-0.011021	*	48.994980
*	2.279906	*	-1.182572	*	-1.198361	*	-0.006142	*	48.945496
*	2.469893	*	-1.228275	*	-1.199309	*	-0.003228	*	48.941635
*	2.659879	*	-1.275705	*	-1.195999	*	0.003239	*	48.983841
*	2.849865	*	-1.264773	*	-1.189154	*	0.016585	*	49.069839
*	3.039851	*	-1.243804	*	-1.180487	*	0.032663	*	49.195465
*	3.229837	*	-1.233457	*	-1.171021	*	0.044216	*	49.355621
*	3.419824	*	-1.199722	*	-1.160979	*	0.050357	*	49.544617
*	3.609810	*	-1.202726	*	-1.150723	*	0.051594	*	49.756378
*	3.799796	*	-1.171800	*	-1.140256	*	0.051657	*	49.984329
*	3.989782	*	-1.172373	*	-1.130201	*	0.051135	*	50.221985
*	4.179769	*	-1.149561	*	-1.120715	*	0.051165	*	50.462906
*	4.369755	*	-1.171543	*	-1.112206	*	0.051478	*	50.701035
*	4.559741	*	-1.166974	*	-1.104602	*	0.054328	*	50.930740
*	4.749727	*	-1.108326	*	-1.098680	*	0.056848	*	51.147034
*	4.939713	*	-1.130142	*	-1.095016	*	0.058735	*	51.346466
*	5.129700	*	-1.045614	*	-1.092589	*	0.056655	*	51.526535
*	5.319686	*	-1.032086	*	-1.092353	*	0.051271	*	51.686035
*	5.509672	*	-1.037953	*	-1.093357	*	0.038513	*	51.825134
*	5.699658	*	-1.055715	*	-1.094423	*	0.026263	*	51.944427
*	5.889645	*	-1.110085	*	-1.095340	*	0.020173	*	52.044235
*	6.079631	*	-1.076871	*	-1.096312	*	0.023103	*	52.124771
*	6.269617	*	-1.110800	*	-1.098415	*	0.027839	*	52.186601
*	6.459603	*	-1.084154	*	-1.101169	*	0.032079	*	52.231140
*	6.649590	*	-1.097795	*	-1.104772	*	0.033428	*	52.260086
*	6.839576	*	-1.086693	*	-1.108665	*	0.032640	*	52.275726
*	7.029562	*	-1.164207	*	-1.112531	*	0.031169	*	52.280350
*	7.219548	*	-11.369934	*	-1.448772	*	-0.003855	*	52.276088
*	7.409534	*	-4.771715	*	-1.065162	*	0.260663	*	52.196152
*	7.599521	*	-4.885085	*	-0.908555	*	0.724280	*	51.901566
*	7.789507	*	-4.941984	*	-0.930544	*	1.104971	*	51.305740
*	7.979493	*	-1.221276	*	-1.154338	*	1.380678	*	50.330307
*	8.169479	*	-1.607866	*	-1.647376	*	1.497520	*	48.969589
*	8.359466	*	-1.866967	*	-2.157928	*	1.178964	*	47.323832
*	8.549452	*	-2.228454	*	-2.533850	*	0.752841	*	45.558022
*	8.739438	*	-2.690789	*	-2.830583	*	0.417827	*	43.956161
*	8.929424	*	-3.164660	*	-3.067161	*	0.236265	*	42.559152

*	9.119411	*	-3.569754	*	-3.222716	*	0.192114	*	41.068508
*	9.309397	*	-3.802303	*	-3.288725	*	0.248152	*	41.537570
*	9.499383	*	-3.889286	*	-3.262568	*	0.364863	*	41.600983
*	9.689369	*	-3.815103	*	-3.148677	*	0.489636	*	42.473740
*	9.879355	*	-3.563250	*	-2.976035	*	0.613066	*	43.858743
*	10.069342	*	-3.206109	*	-2.761302	*	0.718114	*	45.777054
*	10.259328	*	-2.801994	*	-2.526126	*	0.790657	*	47.981018
*	10.449314	*	-2.403330	*	-2.292121	*	0.826747	*	50.369049
*	10.639300	*	-2.053359	*	-2.077678	*	0.831036	*	52.799927
*	10.829287	*	2.769983	*	-2.031552	*	0.837751	*	55.198746
*	11.019273	*	3.203503	*	-1.638245	*	0.107612	*	57.602005
*	11.209259	*	3.445362	*	-1.337570	*	-0.511320	*	60.059918
*	11.399245	*	3.590609	*	-1.067041	*	-0.863214	*	62.670746
*	11.589231	*	-0.761881	*	-0.860477	*	-1.133111	*	65.290519
*	11.779218	*	-0.806800	*	-0.768132	*	-1.088782	*	67.735687

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 3 OF 4

	TIME......(SEC)		STBLTR....(DEG)		AILRON....(DEG)		RUDDER....(DEG)		ALTITUDE...(FT)
*	11.969204	*	-0.915540	*	-0.655329	*	-0.792145	*	70.043457
*	12.159190	*	-0.818488	*	-0.529124	*	-0.461040	*	72.077484
*	12.349176	*	-0.699540	*	-0.417514	*	-0.218913	*	73.742950
*	12.539163	*	-0.612691	*	-0.339986	*	-0.095439	*	74.960114
*	12.729149	*	-0.566952	*	-0.309909	*	-0.054526	*	75.678207
*	12.919135	*	-0.557567	*	-0.324278	*	-0.055930	*	75.877914
*	13.109121	*	-0.575181	*	-0.376057	*	-0.073167	*	75.567383
*	13.299108	*	-0.611483	*	-0.457703	*	-0.093069	*	74.777695
*	13.489094	*	-0.661711	*	-0.561647	*	-0.110690	*	73.558517
*	13.679080	*	-0.739960	*	-0.676090	*	-0.125492	*	71.971045
*	13.869066	*	-0.826459	*	-0.799570	*	-0.138116	*	70.080105
*	14.059052	*	-0.917433	*	-0.928560	*	-0.148069	*	67.978912
*	14.249039	*	-1.008072	*	-1.056622	*	-0.156873	*	65.730789
*	14.439025	*	-1.094344	*	-1.177841	*	-0.162444	*	63.421967
*	14.629011	*	-1.172998	*	-1.287473	*	-0.165273	*	61.128418
*	14.818997	*	-1.241513	*	-1.381750	*	-0.165445	*	58.915155

15.008984	-1.297470	-1.456547	-0.163082	56.856552
15.198970	-1.339688	-1.511010	-0.158429	54.995240
15.388956	-1.367822	-1.545155	-0.151980	53.368027
15.578942	-1.382230	-1.559812	-0.144324	52.005524
15.768929	-1.383876	-1.556629	-0.136056	50.916870
15.958915	-1.374182	-1.537891	-0.127722	50.100784
16.148834	-1.354918	-1.506315	-0.119780	49.544998
16.338730	-1.328072	-1.464860	-0.112589	49.228348
16.528625	-1.295725	-1.416553	-0.106404	49.122620
16.718521	-1.259953	-1.364342	-0.101381	49.154794
16.908417	-1.222731	-1.310954	-0.097588	49.409073
17.098312	-1.185530	-1.257968	-0.094979	49.729767
17.288208	-1.149609	-1.207210	-0.093391	50.123581
17.478104	-1.116189	-1.160531	-0.092715	50.559036
17.667999	-1.086252	-1.119246	-0.092859	51.007065
17.857895	-1.060576	-1.084332	-0.093730	51.441833
18.047791	-1.039686	-1.056404	-0.095221	51.841492
18.237686	-1.023868	-1.035758	-0.097213	52.188431
18.427582	-1.013171	-1.022363	-0.099579	52.469574
18.617477	-1.007424	-1.015905	-0.102191	52.676392
18.807373	-1.006281	-1.015826	-0.104924	52.804011
18.997269	-1.009237	-1.021379	-0.107659	52.853958
19.187164	-1.015683	-1.031667	-0.110293	52.827713
19.377060	3.324049	-1.988774	-0.060459	52.717728
19.566956	-1.054458	-2.290795	-0.521140	52.555893
19.756851	-1.316113	-2.873454	-1.040701	52.734253
19.946747	-1.960548	-3.514855	-1.334141	53.860951
20.136642	-2.005396	-4.166123	-1.408184	56.644333
20.326538	-1.250170	-4.715120	-1.425174	61.550415
20.516434	-0.977447	-5.256442	-1.052974	68.895752
20.706329	-0.714183	-5.551373	-1.111055	76.748398
20.896225	-0.536378	-5.656308	-1.373256	91.015167
21.086121	7.934396	-6.261881	-1.550011	105.502518
21.276016	3.334913	-4.585189	-2.184981	121.881042
21.465912	5.091518	-3.823456	-2.715747	139.351983
21.655807	4.683903	-3.211911	-3.098340	157.130630
21.845703	4.476401	-2.668601	-3.310959	174.301590
22.035599	-0.707323	-2.269772	-3.287128	190.300354
22.225494	-0.402787	-2.197576	-3.446665	204.730820
22.415390	-1.018952	-2.253961	-2.931566	217.343460
22.605286	-0.957839	-2.276559	-2.294298	227.984009
22.795181	-1.466182	-2.234136	-1.783365	236.559372
22.985077	-1.966421	-2.157530	-1.464703	243.024078
23.174973	-2.224481	-2.042295	-1.215720	247.387035
23.364868	-2.933079	-1.874610	-0.999400	249.710815
23.554764	-3.378541	-1.656700	-0.765540	250.106537
23.744659	-4.530933	-1.430122	-0.548162	248.725403
23.934555	-5.335417	-1.202142	-0.331694	245.756470

*	24.124451	*	-4.849390	*	-0.962144	*	-0.114661	*	241.418747

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 3 OF 4

	TIME......(SEC)		STBLTR....(DEG)		AILRON....(DEG)		RUDDER....(DEG)		ALTITUDE...(FT)
*	24.314346	*	-4.733329	*	-0.741563	*	0.101332	*	235.931198
*	24.504242	*	-4.620466	*	-0.578142	*	0.231385	*	229.489410
*	24.694138	*	-4.527917	*	-0.465651	*	0.289164	*	222.255127
*	24.884033	*	-4.474320	*	-0.393226	*	0.322519	*	214.351547
*	25.073929	*	-4.457426	*	-0.346899	*	0.364436	*	205.859080
*	25.263824	*	-4.467105	*	-0.351945	*	0.428672	*	196.827133
*	25.453720	*	4.274610	*	-0.385897	*	0.500631	*	187.284653
*	25.643616	*	-0.232819	*	-0.548033	*	0.558074	*	177.287750
*	25.833511	*	-0.198692	*	-0.756084	*	0.343086	*	166.953583
*	26.023407	*	-0.690840	*	-0.927217	*	0.039595	*	156.390306
*	26.213303	*	0.102798	*	-1.071691	*	-0.150439	*	145.687653
*	26.403198	*	-1.248897	*	-1.217293	*	-0.305899	*	134.929001
*	26.593094	*	-0.333312	*	-1.352807	*	-0.341328	*	124.194458
*	26.782990	*	-1.492764	*	-1.492980	*	-0.385306	*	113.564758
*	26.972885	*	-0.769291	*	-1.620697	*	-0.369920	*	103.123795
*	27.162781	*	-3.682539	*	-1.775579	*	-0.400695	*	92.954556
*	27.352676	*	-4.034352	*	-1.926140	*	-0.148369	*	83.127731
*	27.542572	*	-15.000000	*	0.000000	*	0.402904	*	73.705917
*	27.732468	*	-5.632820	*	-2.184797	*	0.416052	*	64.775589
*	27.922363	*	-6.096888	*	-2.257270	*	0.805972	*	56.263779
*	28.112259	*	-6.717966	*	-2.657307	*	1.135191	*	48.184021
*	28.302155	*	-7.838568	*	-2.994135	*	1.294632	*	40.573837
*	28.492050	*	-9.240601	*	-3.403696	*	1.414780	*	33.465179
*	28.681946	*	-10.259377	*	-3.842335	*	1.650109	*	26.756851
*	28.871841	*	-8.908066	*	-4.308064	*	2.381865	*	20.155396
*	29.061737	*	-1.385618	*	-5.175614	*	3.952714	*	13.361130

CRUISE MISSILE TESTS
BASELINE MISSION SET

GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 4 LF 4

TIME......(SEC)	XM...(FT NORTH)	YM....(FT EAST)	XT...(FT NORTH)	XM....(FT EAST)
* 0.190000	* 159.440155	* -0.009810	*24000.000000	* 6.649995
* 0.380000	* 318.873291	* -0.049694	*24000.000000	* 13.299989
* 0.570000	* 478.301758	* -0.113570	*24000.000000	* 19.949982
* 0.759999	* 637.728271	* -0.201762	*24000.000000	* 26.599976
* 0.949999	* 797.151611	* -0.313580	*24000.000000	* 33.249969
* 1.139989	* 956.573486	* -0.448021	*24000.000000	* 39.899597
* 1.329975	* 1115.991460	* -0.603947	*24000.000000	* 46.549118
* 1.519961	* 1275.407710	* -0.779674	*24000.000000	* 53.198639
* 1.709948	* 1434.820310	* -0.973391	*24000.000000	* 59.848160
* 1.899934	* 1594.228760	* -1.182685	*24000.000000	* 66.497681
* 2.089920	* 1753.632810	* -1.404678	*24000.000000	* 73.147202
* 2.279906	* 1913.032470	* -1.636795	*24000.000000	* 79.796707
* 2.469893	* 2072.426510	* -1.876554	*24000.000000	* 86.446228
* 2.659879	* 2231.814940	* -2.120413	*24000.000000	* 93.095749
* 2.849865	* 2391.197750	* -2.362842	*24000.000000	* 99.745270
* 3.039851	* 2550.574710	* -2.597082	*24000.000000	* 106.394791
* 3.229837	* 2709.944090	* -2.816305	*24000.000000	* 113.044296
* 3.419824	* 2869.307860	* -3.014300	*24000.000000	* 119.693817
* 3.609810	* 3028.665530	* -3.185804	*24000.000000	* 126.343338
* 3.799796	* 3188.016360	* -3.326302	*24000.000000	* 132.992859
* 3.989782	* 3347.361330	* -3.432030	*24000.000000	* 139.642380
* 4.179769	* 3506.700200	* -3.499740	*24000.000000	* 146.291885
* 4.369755	* 3666.032710	* -3.526452	*24000.000000	* 152.941406
* 4.559741	* 3825.359620	* -3.508787	*24000.000000	* 159.590927
* 4.749727	* 3984.681400	* -3.443329	*24000.000000	* 166.240448
* 4.939713	* 4143.984370	* -3.327466	*24000.000000	* 172.889969
* 5.129700	* 4303.257810	* -3.159139	*24000.000000	* 179.539490
* 5.319686	* 4462.531250	* -2.938205	*24000.000000	* 186.188995
* 5.509672	* 4621.804690	* -2.666635	*24000.000000	* 192.838516
* 5.699658	* 4781.078120	* -2.347534	*24000.000000	* 199.488037
* 5.889645	* 4940.351560	* -1.983534	*24000.000000	* 206.137558
* 6.079631	* 5099.625000	* -1.575940	*24000.000000	* 212.787079
* 6.269617	* 5258.898440	* -1.125271	*24000.000000	* 219.436584
* 6.459603	* 5418.171870	* -0.631455	*24000.000000	* 226.086105
* 6.649590	* 5577.445310	* -0.094406	*24000.000000	* 232.735626

*	6.839576	*	5736.703120	*	0.485856	*	24000.000000	*	239.385147
*	7.029562	*	5895.902340	*	1.109634	*	24000.000000	*	246.034008
*	7.219548	*	6055.101560	*	1.785213	*	24000.000000	*	252.684174
*	7.409534	*	6214.300780	*	2.594645	*	24000.000000	*	259.333496
*	7.599521	*	6373.500000	*	3.660096	*	24000.000000	*	265.983154
*	7.789507	*	6532.699220	*	5.144712	*	24000.000000	*	272.632568
*	7.979493	*	6691.898440	*	7.223592	*	24000.000000	*	279.282227
*	8.169479	*	6851.023440	*	10.115357	*	24000.000000	*	285.931641
*	8.359466	*	7010.148440	*	14.135207	*	24000.000000	*	292.581055
*	8.549452	*	7169.210940	*	19.605545	*	24000.000000	*	299.230713
*	8.739438	*	7328.160160	*	26.823135	*	24000.000000	*	305.880127
*	8.929424	*	7486.953120	*	36.051483	*	24000.000000	*	312.529785
*	9.119411	*	7645.527340	*	47.457543	*	24000.000000	*	319.179199
*	9.309397	*	7803.820310	*	61.294495	*	24000.000000	*	325.828857
*	9.499383	*	7961.785160	*	77.494766	*	24000.000000	*	332.478271
*	9.689369	*	8119.371090	*	96.073334	*	24000.000000	*	339.127686
*	9.879355	*	8276.554690	*	116.938950	*	24000.000000	*	345.777344
*	10.069342	*	8433.335940	*	139.948761	*	24000.000000	*	352.426758
*	10.259328	*	8589.730470	*	164.924744	*	24000.000000	*	359.076416
*	10.449314	*	8745.746090	*	191.672058	*	24000.000000	*	365.725830
*	10.639300	*	8901.429690	*	219.956429	*	24000.000000	*	372.375488
*	10.829287	*	9056.800780	*	249.667542	*	24000.000000	*	379.024902
*	11.019273	*	9211.929690	*	280.360840	*	24000.000000	*	385.674316
*	11.209259	*	9366.890620	*	311.764160	*	24000.000000	*	392.323975
*	11.399245	*	9521.742190	*	343.555176	*	24000.000000	*	398.973389
*	11.589231	*	9676.562500	*	375.463623	*	24000.000000	*	405.623047
*	11.779218	*	9831.382810	*	407.341797	*	24000.000000	*	412.272461

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FRECUENCY= 0.20

CATA SET NUMBER 4 OF 4

TIME......(SEC)		XM...(FT NORTH)		YM....(FT EAST)		XT...(FT NORTH)		XM....(FT EAST)	
*	11.969204	*	9986.203120	*	439.127930	*	24000.000000	*	418.922119
*	12.159190	*	10141.023430	*	470.795654	*	24000.000000	*	425.571533
*	12.349176	*	10295.843700	*	502.336670	*	24000.000000	*	432.220947
*	12.539163	*	10450.663600	*	533.751709	*	24000.000000	*	438.870605

*	12.729149	*10605.578100	*	565.044678	*24000.000000	*	445.520020
*	12.919135	*10760.472700	*	596.219727	*24000.000000	*	452.169678
*	13.109121	*10915.398400	*	627.277344	*24000.000000	*	458.819092
*	13.299108	*11070.367200	*	658.218750	*24000.000000	*	465.468750
*	13.489094	*11225.335900	*	689.041992	*24000.000000	*	472.118164
*	13.679080	*11380.375000	*	719.744141	*24000.000000	*	478.767578
*	13.869066	*11535.418000	*	750.322754	*24000.000000	*	485.417236
*	14.059052	*11690.523400	*	780.773926	*24000.000000	*	492.066650
*	14.249039	*11845.640600	*	811.093506	*24000.000000	*	498.716309
*	14.439025	*12000.828100	*	841.276611	*24000.000000	*	505.365723
*	14.629011	*12156.027300	*	871.318848	*24000.000000	*	512.015381
*	14.818997	*12311.293000	*	901.215088	*24000.000000	*	518.664795
*	15.008984	*12466.574200	*	930.961670	*24000.000000	*	525.314209
*	15.198970	*12621.914100	*	960.554443	*24000.000000	*	531.963867
*	15.388956	*12777.261700	*	989.990479	*24000.000000	*	538.613281
*	15.578942	*12932.675800	*	1019.267580	*24000.000000	*	545.262939
*	15.768929	*13088.089800	*	1048.383790	*24000.000000	*	551.912354
*	15.958915	*13243.550800	*	1077.338620	*24000.000000	*	558.562012
*	16.148834	*13399.039100	*	1106.131350	*24000.000000	*	565.208984
*	16.338730	*13554.527300	*	1134.763180	*24000.000000	*	571.855469
*	16.528625	*13710.054700	*	1163.234620	*24000.000000	*	578.501709
*	16.718521	*13865.617200	*	1191.547360	*24000.000000	*	585.148193
*	16.908417	*14021.179700	*	1219.703370	*24000.000000	*	591.794434
*	17.098312	*14176.742200	*	1247.704100	*24000.000000	*	598.440918
*	17.288208	*14332.351600	*	1275.552250	*24000.000000	*	605.087158
*	17.478104	*14487.988300	*	1303.249510	*24000.000000	*	611.733398
*	17.667999	*14643.625000	*	1330.797610	*24000.000000	*	618.379883
*	17.857895	*14799.261700	*	1358.198240	*24000.000000	*	625.026123
*	18.047791	*14954.949200	*	1385.453370	*24000.000000	*	631.672607
*	18.237686	*15110.660200	*	1412.564210	*24000.000000	*	638.318848
*	18.427582	*15266.371100	*	1439.531740	*24000.000000	*	644.965332
*	18.617477	*15422.082000	*	1466.356450	*24000.000000	*	651.611572
*	18.807373	*15577.867200	*	1493.039310	*24000.000000	*	658.258057
*	18.997269	*15733.652300	*	1519.580080	*24000.000000	*	664.904297
*	19.187164	*15889.437500	*	1545.979250	*24000.000000	*	671.550537
*	19.377060	*16045.246100	*	1572.180910	*24000.000000	*	678.197021
*	19.566956	*16201.109400	*	1598.057130	*24000.000000	*	684.843262
*	19.756851	*16357.054700	*	1623.358890	*24000.000000	*	691.489746
*	19.946747	*16513.078100	*	1647.744380	*24000.000000	*	698.135986
*	20.136642	*16669.160200	*	1670.831050	*24000.000000	*	704.782471
*	20.326538	*16825.242200	*	1692.221190	*24000.000000	*	711.428711
*	20.516434	*16981.269500	*	1711.564210	*24000.000000	*	718.074951
*	20.706329	*17137.136700	*	1728.602290	*24000.000000	*	724.721436
*	20.896225	*17292.789100	*	1743.127440	*24000.000000	*	731.367676
*	21.086121	*17448.156200	*	1754.952150	*24000.000000	*	738.014160
*	21.276016	*17603.203100	*	1763.747310	*24000.000000	*	744.660400
*	21.465912	*17757.972700	*	1769.209720	*24000.000000	*	751.306885
*	21.655807	*17912.515600	*	1771.405030	*24000.000000	*	757.953125

*	21.845703	*18066.910200	*	1770.707280	*24000.000000	*	764.599609
*	22.035599	*18221.203100	*	1767.581790	*24000.000000	*	771.245850
*	22.225494	*18375.429700	*	1762.431400	*24000.000000	*	777.892090
*	22.415390	*18529.609400	*	1755.413820	*24000.000000	*	784.538574
*	22.605286	*18683.722700	*	1746.558110	*24000.000000	*	791.184814
*	22.795181	*18837.746100	*	1735.896970	*24000.000000	*	797.831299
*	22.985077	*18991.644500	*	1723.493900	*24000.000000	*	804.477539
*	23.174973	*19145.402300	*	1709.431880	*24000.000000	*	811.124023
*	23.364868	*19299.007800	*	1693.809570	*24000.000000	*	817.770264
*	23.554764	*19452.437500	*	1676.730710	*24000.000000	*	824.416504
*	23.744659	*19605.695300	*	1658.295170	*24000.000000	*	831.062988
*	23.934555	*19758.785200	*	1638.600340	*24000.000000	*	837.709229
*	24.124451	*19911.699200	*	1617.746830	*24000.000000	*	844.355713

CRUISE MISSILE TESTS
BASELINE MISSION SET
GLINT PLUS ECM AT 0.2 HZ
9-19-84

*** BLINKER FREQUENCY= 0.20

DATA SET NUMBER 4 OF 4

TIME......(SEC)		XM...(FT NORTH)		YM....(FT EAST)	XT...(FT NORTH)		XM....(FT EAST)
*	24.314346	*20064.460900	*	1595.848880	*24000.000000	*	851.001953
*	24.504242	*20217.089800	*	1573.031490	*24000.000000	*	857.648437
*	24.694138	*20369.617200	*	1549.432370	*24000.000000	*	864.294676
*	24.884033	*20522.062500	*	1525.158730	*24000.000000	*	870.941162
*	25.073929	*20674.441400	*	1500.484370	*24000.000000	*	877.587402
*	25.263824	*20826.812500	*	1475.440190	*24000.000000	*	884.233643
*	25.453720	*20979.183600	*	1450.201660	*24000.000000	*	890.880127
*	25.643616	*21131.578100	*	1424.848630	*24000.000000	*	897.526367
*	25.833511	*21284.023400	*	1399.423830	*24000.000000	*	904.172852
*	26.023407	*21436.527300	*	1373.927490	*24000.000000	*	910.819092
*	26.213303	*21589.078100	*	1348.337400	*24000.000000	*	917.465576
*	26.403198	*21741.671900	*	1322.615970	*24000.000000	*	924.111816
*	26.593094	*21894.339800	*	1296.726810	*24000.000000	*	930.758057
*	26.782990	*22047.039100	*	1270.628660	*24000.000000	*	937.404541
*	26.972885	*22199.781200	*	1244.285160	*24000.000000	*	944.050781
*	27.162781	*22352.566400	*	1217.680910	*24000.000000	*	950.697266
*	27.352676	*22505.390600	*	1190.884520	*24000.000000	*	957.343506
*	27.542572	*22658.285200	*	1164.029050	*24000.000000	*	963.989990

*	27.732468	*	22811.261700	*	1137.282230	*	24000.000000	*	970.636230
*	27.922363	*	22964.343700	*	1110.862550	*	24000.000000	*	977.282715
*	28.112259	*	23117.585900	*	1085.108890	*	24000.000000	*	983.928955
*	28.302155	*	23271.027300	*	1060.469970	*	24000.000000	*	990.575195
*	28.492050	*	23424.718700	*	1037.558840	*	24000.000000	*	997.221680
*	28.681946	*	23578.683600	*	1017.097900	*	24000.000000	*	1003.867920
*	28.871841	*	23732.902300	*	999.782715	*	24000.000000	*	1010.514400
*	29.061737	*	23887.324200	*	986.245850	*	24000.000000	*	1017.160640

APPENDIX C

TASM SIMULATION PROGRAM NOMENCLATURE

```
***************************************************************************

CONTROL

  TIME                   TIME IN SECONDS
  DT                     INTEGRATION INTERVAL
  OPCT                   OUTPUT INTERVAL
  NPTS                   NUMBER OF OUTPUT SETS SAVED
  CPA                    FLAG SET TO INDICATE TERMINATION & REASON
  PTS                    ARRAY CONTAINING ALL SAVED DATA
  NFAZE                  SETS THE PHASE APPLIED TO ECM BLINKER
  PCFRNG                 SETS THE RANGE AT WHICH CLIMB IS COMMENCED

MISSILE DYNAMICS

  U,V,W                  BODY AXIS LINEAR VELOCITIES (FT/SEC)
  UDOT,VDOT,WDOT         BODY AXIS LINEAR ACCELERATIONS
  P,Q,R                  BODY AXIS ANGULAR VELOCITIES (RAD/SEC)
  ROLLRT,PTCHRT,YAWRT             "                 (DEG/SEC)
  PDOT,QDOT,RDOT         BODY AXIS ANGULAR ACCELERATIONS
  X,Y,Z                  BODY AXIS AERODYNAMIC FORCES (LBS)
  L,D                    LIFT, DRAG AERODYNAMIC FORCES (LBS)
  LA,MA,NA               BODY AXIS AERODYNAMIC MOMENTS (FT-LBS)
  PHI,THETA,SY           EULER ANGLES (RAD)
  BANK,PITCH,HEADNG           "      (DEG)
  PHIDOT,THETAD,SYDOT         "      RATE OF CHANGE OF
  ALFA,BETA              ANGLE OF ATTACK, SIDESLIP (RAD)
  AOA,SIDESL                  "             "    (DEG)
  ALFADT,BETADT          RATE OF CHANGE OF ALFA,BETA
  GAMMA                  FLIGHT PATH ANGLE (RAD)
  FLTPTH                      "            (DEG)
  NY,NZ                  LATERAL ACCELERATION, LOAD FACTOR (G'S)
  XM,YM,ALTUDE           EARTH COORDINATES OF MISSILE (FT)
                         (XM-NORTH,YM-EAST)
  XMDOT,YMDOT,HMDOT      RATES OF CHANGE OF XM, YM, ALTITUDE
  VT                     TOTAL MISSILE VELOCITY (FT/SEC)
  CHORD,SPAN             MEAN AERODYNAMIC CHORD, SPAN (FT)
  CHORD2, SPAN2          HALF CHORD, HALF SPAN
```

```
  WT,M,G              GROSS WEIGHT, MASS, ACCEL DUE TO GRAVITY
  T,S                 THRUST, WING AREA
  RHO                 AIR DENSITY
  QS                  DYNAMIC PRESSURE X WING AREA
  C--                 AERODYNAMIC COEFFICIENTS
  DC---               INCREMENTS IN     "
  IXX,IYY,IZZ,IXZ     MOMENTS AND PRODUCTS OF INERTIA
  IA - IK             FUNCTIONS OF      "
  ELE,AIL,RUD         STANDARD CONTROL DEFLECTIONS (DEG)
  STBLTR,AILRON,RUDDER          CONTROL DEFLECTIONS WITH LIMITS
                                APPLIED (DEG)
  RSTAB1,LSTAB1       TASM UNLIMITED CONTROL DEFLECTIONS (DEG)
  RSTAB,LSTAB         TASM LIMITED CONTROL DEFLECTIONS (DEG)
  ---1                INITIAL CONDITION

AUTOPILOT

  K---                AUTOPILOT GAINS
  CGARM-              ACCELEROMETER LOCATION WRT CG
  E---                COMPARATOR ERRORS
  ---LIM              LIMITED VALUES
  ---SERI             SERVO INPUTS
  ---SERO             SERVO OUTPUTS
  ---C                COMMANDED VALUES
  ---F,---FF.---FFF   FILTERED SENSOR VARIABLE
  FNCOF-,FDCOF-       NOTCH FILTER COEFFICIENTS

GUIDANCE

  AZC,AYC             COMMANDED VERTICAL AND HORIZONTAL
                      ACCELERATIONS IN EARTH AXES (G'S)
  XT,YT,HT            EARTH COORDINATES OF TARGET (FT)
  XR,YR,HR            POSITION OF THE TARGET WRT MISSILE IN
                      EARTH COORDINATES (FT)
  XRECM,YRECM,HRECM   POSITION OF RADAR TARGET (XR,YR,HR WITH ECM
                      AND GLINT ADDED WRT MISSILE (FT)
  XDOTR,YDOTR,HDOTR   RATES OF CHANGE OF XRECM,YRECM,HRECM (FT/S)
                      (RELATIVE VELOCITY OF TGT WRT MISSILE)
  XECM,YECM,HECM      INCR. IN TARGET RADAR POSITION DUE TO ECM
  XGLNT,YGLNT,HGLNT   INCR. IN TARGET RADAR POSITION DUE TO GLINT
  RANGE               RANGE TO TARGET (FT)
  RNGECM,RGECMT       RANGE TO RADAR TARGET (TGT WITH ECM+GLINT)
  TSPEED              TARGET SPEED (FT/SEC)
  SYT,THETAT          HEADING, ELEV. TO TARGET FROM MISSILE (RAD)
  HEADT,ELEVT         HEADING, ELEV. TO TARGET FROM MISSILE (DEG)
  VTANAZ,VTANEL       COMPONENT OF RELATIVE VEL. PERPENDICULAR
                      TO LOS IN AZIMUTH AND ELEVATION (FT/SEC)
  TRAKAZ,TRAKEL       COMPONENT OF MISSILE VELOCITY VECTOR IN
```

	AZIMUTH AND ELEVATION (FT/SEC)
SIGAZ,SIGEL	EARTH AZIMUTH, ELEVATION LOS ANGLES (RAD)
DSIGAZ,DSIGEL	EARTH AZIMUTH, ELEVATION LOS ANGLES (DEG)
SIGDAZ,SIGDEL	RATES OF CHANGE OF EARTH REFERENCED LOS (RAD/SEC)
DSGDAZ,DSGDEL	SAME AS ABOVE (DEG/SEC)
SIGDAF,SIGDEF	FILTERED SIGDAT,SIGDET (RAD/SEC)
DSGDAF,DSGDEF	FILTERED DSGDAT,DSGDET (DEG/SEC)
LAMDAZ,LAMDEL	PROPORTIONAL NAVIGATION CONSTANTS
FREQ	FREQUENCY OF ECM BLINKING
SHIFTY,SHIFTH	DISTANCE OF ECM BLINKER FROM TARGET AIM POINT (FT)
BRNTHR	BURN-THROUGH RANGE
KNFAZ,KNFEL	AZIMUTH AND ELEVATION NAVIGATION FILTER CONSTANTS

```
C
C
C
C                              APPENDIX D
C
C                         MAIN PROGRAM LISTING
C                                 FOR
C                  TACTICAL CRUISE MISSILE SIMULATION
C
C                                 ***
C                            TRANSLATED BY
C                   CDR BARTON P. ANDERSON, USN
                                  ***
C                      NAVY POSTGRADUATE SCHOOL
C            DEPARTMENT OF AERONAUTICAL ENGINEERING
C                        MONTEREY, CA 93943
C                                 ***
C                  TRANSLATED FROM CSMP PROGRAM BY
C                          DR. MARLE HEWETT
C                      LCDR KENT WATTERSON, USN
C
C                                                                    9-10-84
C *********************************************************************
C     PROGRAM TCMC
C *********************************************************************
C     CONTROLS THE OVERALL EXECUTION OF THE SIMULATION. CALLS THE
C     NECESSARY SUBROUTINES, DETERMINES WHEN THE DATA MUST BE STORED
C     FOR OUTPUT AND WHEN THE RUN HAS COMPLETED DUE TO CPA OR FINTIM.
C     *****************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOUT,PCOUNT,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     ******************************************
C
      COMMON /A/ TIME  ,FINTIM,DT    ,CPDT ,NOUT ,NPTS ,CPA   ,PCOUNT,
     *           G     ,T     ,RHO   ,PI   ,PII  ,MASS ,WT    ,S     ,
     *           IXX   ,IYY   ,IZZ   ,IXZ  ,IA   ,IB   ,IC    ,CS    ,
     *           ID    ,IE    ,IF    ,IG   ,IH   ,II   ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2 ,SPAN ,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /H/: ECM/GLINT PARAMETERS
C     ******************************************
```

```
C
      COMMON /H/ FREQ           ,SHIFTY         ,SHIFTH         ,BRNTHR        ,
     *           XECM           ,YECM           ,HECM           ,XGLNT         ,
     *           YGLNT          ,HGLNT          ,XTECM          ,YTECM         ,
     *           HTECM
C
C     ******************************************
C     COMMON BLOCK /F/: GUIDANCE PARAMETERS
C     ******************************************
C
      COMMON /F/ PH1            ,PH2            ,PH3            ,PH4           ,
     *           OFFSET         ,ALTATT         ,SGDZPU         ,MISDST        ,
     *           LAMDAZ         ,LAMDEL         ,KNFAZ          ,KNFEL         ,
     *           NZC            , PHIC          ,GAMMAC         ,PCLIM         ,
     *           PC             ,QC             ,RC             ,RANGE         ,
     *           SIGAZ          ,SIGEL          ,SIGDAF         ,SIGDEF        ,
     *           SYT            ,THETAT         ,XT             ,YT            ,
     *           HT             ,NYC            ,POPRNG
C
C
C     ******************************************
C     COMMON BLOCK /I/: OUTPUT ARRAY
C     ******************************************
C
      COMMON /I/ PTS(300,20)
C
C*********************************************************************
C
C *** INITIALIZE ALL VARIABLES
C
C     PLOTTING SURFACE
C
C
      CALL TEK618
      CALL COMPRS
      CALL SWISSM
C
C
    1 CALL INIT
C
C *** BEGIN DYNAMIC SIMULATION ***************************************
C
C
   10 IF(CPA.GE.1.0) GO TO 100
            TIME=TIME+DT
            IF(TIME.GE.FINTIM) CPA=2.0
            PCOUNT =   PCOUNT+1
C
C
C *** FROM MISSION PROFILE/GUIDANCE, GENERATE NZC,PHIC
```

```
C
            CALL MISSN1
C
C
C *** GENERATE CONTROL MOVEMENTS: STBLTR, AILRON, RUDDER
C
            CALL APILOT
C
C
C *** GENERATE MISSILE MOTION AND POSITION
C
            CALL AERO
C
C
C *** GENERATE APPARENT RADAR TARGET POSITION & MOTION
C
            CALL TGTNAV
C
C
C *** STORE REQUIRED PLOT DATA IN THE PTS ARRAY.
C
            IF(PCOUNT.LT.NOUT)GO TO 50
                CALL PREPAR
                PCOUNT = 0.0
  50        CONTINUE
C
            GO TO 10
C
 100 CONTINUE
C
C *** END DYNAMIC SIMULATION ******************************************
C
C
C *** INVOKE DISSPLA AND TABULAR OUTPUT ROUTINES
C
      CALL OUTPUT(NPTS,CPA)
C
C ************* TESTPATCH TO BYPASS ITERATIONS (KTEST = 1)
      KTEST = 0
      IF(KTEST.EQ.1) GO TO 150
C ************* ENDTEST
C
C *** ITERATE THE PHASE VARIABLE IN THE ECM PACKAGE
C
      NFAZE  = NFAZE + 1
      IF(NFAZE.LE.4)GO TO 1
      NFAZE = 1
C
```

```
C *** ITERATE THE BLINKER FREQUENCY
C
      FREQ = FREQ +5.0
C
      IF(FREQ.LE.30.0)  GO TO 1
 150  CALL DONEPL
C
      STOP
      END
C
C                                                                    9-07-84
C*****************************************************************************
      SUBROUTINE INIT
C*****************************************************************************
C     BLOCK DATA
C************************
C     CONTAINS DATA STATEMENTS AND ASSIGNMENT STATEMENTS FOR
C     VARIABLES NOT INITIALIZED IN THE BLOCK DATA SUBPROGRAM BELOW.
C     ALSO REINITIALIZES INITIAL CONDITIONS FOR MORE THAN ONE RUN.
C     ***************************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOUT,PCOUNT,NFAZE
C
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPDT  ,NOUT  ,NPTS  ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI    ,PII   ,MASS  ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ   ,IA    ,IB    ,IC    ,QS    ,
     *           ID     ,IE    ,IF    ,IG    ,IH    ,II    ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2,SPAN  ,NFAZE
C
      COMMON /C/ KPTCHR        ,KROLLR        ,KYAWRT        ,KBANK         ,
     *           KGAMMA        ,KALT          ,GGARML        ,GGARMN        ,
     *           KRTLIM        ,PLIM          ,KNY           ,KNZ           ,
     *           AILRON        ,STBLTR        ,RUDDER        ,
     *           BSERO         ,NZSERO        ,NYSERO
C
      COMMON /D/ ALFA          ,BETA          ,VT            ,HMDOT         ,
     *           U             ,V             ,W             ,
     *           PHI           ,GAMMA         ,THETA         ,SY            ,
     *           CD            ,CY            ,CL            ,CR            ,
     *           CM            ,CN            ,P             ,Q             ,
     *           R             ,ALFADT        ,BETADT        ,PDOT          ,
     *           QDOT          ,RDOT          ,NZ            ,ALTUDE        ,
     *           XM            ,YM            ,XMDOT         ,YMDOT
C
      COMMON /E/ XM1           ,YM1           ,ALTUD1        ,U1            ,
     *           PITCH1        ,BANK1         ,HEDNG1        ,SIDES1        ,
```

```
     *              ADA1             ,PTCHR1          ,ROLLR1          ,YAWR1          ,
     *              XT1              ,YT1             ,HT1             ,TSPEED
C
      COMMON /F/ PH1                 ,PH2             ,PH3             ,PH4            ,
     *              OFFSET           ,ALTATT          ,SGD2PU          ,MISDST         ,
     *              LAMDAZ           ,LAMDEL          ,KNFAZ           ,KNFEL          ,
     *              NZC              , PHIC           ,GAMMAC          ,PCLIM          ,
     *              PC               ,QC              ,RC              ,RANGE          ,
     *              SIGAZ            ,SIGEL           ,SIGDAF          ,SIGDEF         ,
     *              SYT              ,THETAT          ,XT              ,YT             ,
     *              HT               ,NYC             ,PUPRNG
C
      COMMON /H/ FREQ                ,SHIFTY          ,SHIFTH          ,BRNTHR         ,
     *              XECM             ,YECM            ,HECM            ,XGLNT          ,
     *              YGLNT            ,HGLNT           ,XTECM           ,YTECM          ,
     *              HTECM
C
      COMMON /I/ PTS(300,20)
C
C
C EXECUTABLE STMTS ***************************************************************
C
C *** COMPUTED CONSTANTS
C
      PII= 180.0/PI
      IA  =  IXX*IZZ-IXZ**2
      IB  =  IZZ/IA
      IC  =  IXZ/IA
      ID  =  IXZ*(IYY-IXX-IZZ)/IA
      IE  =  (IZZ**2-IYY*IZZ+IXZ**2)/IA
      IF  =  1/IYY
      IG  =  (IXX-IZZ)/IYY
      IH  =  IXZ/IYY
      II  =  IXX/IA
      IJ  =  (IXX*IYY-IXX**2-IXZ**2)/IA
      IK  =  (IXX+IZZ-IYY)*IXZ/IA
C
      PLIM    = RRTLIM/PII
C
C ******************************************************************************
C     PARAMETERS TO BE INITIALIZED FOR EACH RUN FOLLOW.
C     *************************************************************************
C
      TIME = 0.0
      CPA  = 0
      PCOUNT     = 0
      NOUT = INT(CPDT/DT)
      IF(AMOD(CPDT,DT).GE.0.5) NOUT=NOUT+1
```

```
C
      NPTS = C
      DU 100 I=1,300
            DC 50 J=1,20
                  PTS(I,J)   = 0.0
   5C       CCNTINLE
  10C CUNTINLE
C
      U  =  L1
      W  =  L1*TAN(AOA1/PII)
      V  =  SQRT(U1**2+W**2)*TAN(SIDES1/PII)
      VT = SQRT(U**2+V**2+W**2)
      QS = 12.0*RHO*(VT**2)/2.0
C
      P  =  RCLLR1/PII
      Q  =  FTCHR1/PII
      R  =  YAWR1/PII
C
      PDCT    = 0.0
      QDCT    = 0.0
      RDOT    = 0.0
C
      THETA   = PITCH1/PII
      PHI     = BANK1/PII
      SY      = HEDNG1/PII
      GAMMA   = 0.0
      NZ      = 1.0
C
C
      XM    = XM1
      YM    = YM1
      ALTUDE      = ALTUD1
C
      AOA   = ACA1
      ALFA    = ACA1/PII
      ALFADT      = 0.0
C
      SICESL      = SIDES1
      BETA    = SIDES1/PII
      BETADT      = 0.0
C
      XT    = XT1
      YT    = YT1
      HT    = HT1
      RANGE  = SQRT((XT-XM)**2+(YT-YM)**2+(HT-ALTUDE)**2)
C
      PH1   = 0
      PH2   = C
```

```
      PH3   = 0
      PH4   = 0
C
C *** INITIAL CONTROL POSITIONS
C
      ELE1    = -1.0
      AIL1    =  0.0
      RUD1    =  0.0
C
      NZSERO = -ELE1/PII
      BSERO  = -AIL1/PII
      NYSERO = -RUD1/PII
C
C *** INITIAL ACCELERATIONS & COMMANDED RATES
C
      NZ      = 1.0
      NY      = 0.0
C
      PC    = 0.0
      QC    = -ELE1/PII
      RC    = 0.0
C
C *** PARAMETERS TO BE CHANGED FROM BASELINE VALUES
C
C
      FINTIM = 30.0
      KRCLLR = 0.5
      ALTATT = 0.0
      BRNTHR= 500
C
      SIGDAF = TSPEED/XT1
      SIGDEF = (HT1-ALTUD1)*U1/XT1**2
C
      RETURN
      END
C
C
C
C
C                                                                     9-07-84
C ***************************************************************************
C     BLOCK DATA SUBPROGRAM TO INITIALIZE COMMON BLOCKS OF MEMORY.
C ***************************************************************************
      BLOCK DATA
C     **********
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,NPTS,CPA,NOUT,PCOUNT,TITLE,PLTN,LEG,XN,
```

```
     *         YN,NFAZE
C
      COMMON /A/ TIME    ,FINTIM,DT    ,OPDT  ,NCUT  ,NPTS  ,CPA   ,PCOUNT,
     *           G       ,T     ,RHO   ,PI    ,PII   ,MASS  ,WT    ,S     ,
     *           IXX     ,IYY   ,IZZ   ,IXZ   ,IA    ,IB    ,IC    ,QS    ,
     *           ID      ,IE    ,IF    ,IG    ,IH    ,II    ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2,SPAN  ,NFAZE
C
      COMMON /B/ LFT1(2,36)  ,LFT2(2,36)  ,DRG1(2,36)  ,DRG2(2,36)  ,
     *           DRG3(2,36)  ,DRG4(2,36)  ,PTCH1(2,36) ,PTCH2(2,36) ,
     *           SID1(6,10)  ,SID2(6,10)  ,SID3(6,10)  ,DREC1(6,10) ,
     *           DREC2(6,10) ,DREC3(6,10) ,LTRL1(6,10) ,LTRL2(6,10) ,
     *           LTRL3(6,10) ,CLADT       ,CDADT       ,CMADT       ,
     *           CLQ         ,CDQ         ,CMQ         ,CRBDT       ,
     *           CYR         ,CNBDT       ,CRR         ,CYP         ,
     *           CNR         ,CRP         ,CYBDT       ,CNP
C
      COMMON /C/ KPTCHR      ,KROLLR      ,KYAWRT      ,KBANK       ,
     *           KGAMMA      ,KALT        ,CGARML      ,CGARMN      ,
     *           RRTLIM      ,PLIM        ,KNY         ,KNZ         ,
     *           AILRON      ,STBLTR      ,RUDDER      ,
     *           BSERO       ,NZSERO      ,NYSERO
C
      COMMON /D/ ALFA        ,BETA        ,VT          ,HMDOT       ,
     *           U           ,V           ,W           ,
     *           PHI         ,GAMMA       ,THETA       ,SY          ,
     *           CD          ,CY          ,CL          ,CR          ,
     *           CM          ,CN          ,P           ,Q           ,
     *           R           ,ALFADT      ,BETADT      ,PDOT        ,
     *           QDOT        ,RDOT        ,NZ          ,ALTUDE      ,
     *           XM          ,YM          ,XMDOT       ,YMDOT
C
      COMMON /E/ XM1         ,YM1         ,ALTUD1      ,U1          ,
     *           PITCH1      ,BANK1       ,HEDNG1      ,SIDES1      ,
     *           AOA1        ,PTCHR1      ,ROLLR1      ,YAWR1       ,
     *           XT1         ,YT1         ,HT1         ,TSPEED
C
      COMMON /F/ PH1         ,PH2         ,PH3         ,PH4         ,
     *           OFFSET      ,ALTATT      ,SGDZPO      ,MISDST      ,
     *           LAMDAZ      ,LAMDEL      ,KNFAZ       ,KNFEL       ,
     *           NZC         , PHIC       ,GAMMAC      ,PCLIM       ,
     *           PC          ,QC          ,RC          ,RANGE       ,
     *           SIGAZ       ,SIGEL       ,SIGDAF      ,SIGDEF      ,
     *           SYT         ,THETAT      ,XT          ,YT          ,
     *           HT          ,NYC         ,PUPRNG
C
      COMMON /H/ FREQ        ,SHIFTY      ,SHIFTH      ,BRNTHR      ,
     *           XECM        ,YECM        ,HECM        ,XGLNT       ,
```

```
     *              YGLNT         ,HGLNT        ,XTECM        ,YTECM        ,
     *              FTECM
C
C
C      ******************************************
C      COMMON BLOCK /I/: OUTPUT ARRAY
C      ******************************************
C
       COMMON /I/ PTS(300,20),PLTN(6,7),XN(6,7),YN(6,7),TITLE(8,4),
     *            LEG(4,20)
C
C
C
C*** INITIALIZE THE DATA
C
C      ******************************************
C      COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C      ******************************************
C
C
       DATA        TIME  ,FINTIM ,DT      ,OPDT      ,CPA
     *            /0.0   ;30.0   ;0.01    ;0.20      ;0       /
C
       DATA        G     ,T      ,RHO     ,PI        ,WT      ,MASS
     *            /32.17 ;242.3  ;.002377 ;3.141593  ;2200.   ;68.38   /
C
       DATA        IXX   ,IYY    ,IZZ     ,IXZ       ,S
     *            /27.8  ;1507.0 ;1512.0  ;11.7      ;12.0    /
C
       DATA        CHORD2,CHORD  ,SPAN2   ,SPAN      ,NFAZE
     *            /0.707 ;1.414  ;4.2425  ;8.485     ;1        /
C
C
C
C      ******************************************
C      COMMON BLOCK /B/: AERODYNAMIC COEFFICIENT TABLES
C      ******************************************
C
C
C ** STATIC AERODYNAMIC COEFFICIENTS
C
C   * 1.  LIFT COEFFICIENT DATA
C
C          A.  CLBAS VS. AOA (BASIC LIFT COEFFICIENT AS A FUNCTION
C              OF ANGLE OF ATTACK)
C
       DATA LFT1/ -12.0,-0.32,  -11.0,-0.39,  -10.0,-0.46,  -9.0,-0.57,
     *             -8.0,-0.70,   -7.0,-0.69,   -6.0,-0.65,  -5.0,-0.58,
```

```
     *         -4.0,-0.50,     -3.0,-0.42,     -2.0,-0.30,     -1.0,-0.16,
     *          0.0,-0.08,      1.0, 0.04,      2.0, 0.15,      3.0, 0.25,
     *          4.0, 0.36,      5.0, 0.47,      6.0, 0.58,      7.0, 0.69,
     *          8.0, 0.80,      9.0, 0.87,     10.0, 0.78,     11.0, 0.70,
     *         12.0, 0.64,     22*9999.0/
C
C
C        B.   DCLSTE VS. STBLTR (INCREMENT IN LIFT COEFFICIENT
C             DUE TO SYMMETRIC STABILATOR DEFLECTION)
C
      DATA LFT2/ -15.0,-.100,  -14.0,-.097,  -13.0,-.094,  -12.0,-.090,
     *          -11.0,-.084,  -10.0,-.078,   -9.0,-.071,   -8.0,-.065,
     *           -7.0,-.056,   -6.0,-.048,   -5.0,-.040,   -4.0,-.032,
     *           -3.0,-.024,   -2.0,-.016,   -1.0,-.008,    0.0, .000,
     *            1.0, .008,    2.0, .016,    3.0, .024,    4.0, .033,
     *            5.0, .041,    6.0, .049,    7.0, .057,    8.0, .065,
     *            9.0, .073,   10.0, .080,   11.0, .086,   12.0, .092,
     *           13.0, .096,   14.0, .098,   15.0, .100,   10*9999./
C
C
C    *  2.   DRAG COEFFICIENT DATA
C
C        A.   CDBAS VS. CLBAS (BASIC DRAG COEFFICIENT AS A FUNCTION
C             OF BASIC LIFT COEFFICIENT)
C
      DATA DRG1/  -.9, .080,    -.8, .061,    -.7, .050,    -.6, .042,
     *            -.5, .035,    -.4, .031,    -.3, .026,    -.2, .023,
     *            -.1, .022,    0.0, .022,    0.1, .023,    0.2, .024,
     *            0.3, .026,    0.4, .031,    0.5, .036,    0.6, .042,
     *            0.7, .049,    0.8, .057,    0.9, .068,    1.0, .080,
     *            32*9999./
C
C
C        B.   CDDSTE VS. STBLTR (INCREMENT IN DRAG COEFFICIENT
C             DUE TO SYMMETRIC STABILATOR DEFLECTION)
C
      DATA DRG2/ -15., .0106,  -14., .0091,  -13., .0077,  -12., .0065,
     *          -10., .0044,   -9., .0036,   -8., .0028,   -7., .0022,
     *           -6., .0016,   -5., .0011,   -4., .0007,   -3., .0003,
     *           -2., .0001,   -1., 0.   ,    0., 0.   ,    1., .0001,
     *            2., .0004,    3., .0007,    4., .0011,    5., .0016,
     *            6., .0023,    7., .0030,    8., .0039,    9., .0049,
     *           10., .0060,   11., .0073,   12., .0089,   13., .0106,
     *           14., .0126,   15., .0146,   12*9999.  /
C
C        C.   CDDSTA VS. AILRON (INCREMENT IN DRAG COEFFICIENT
C             DUE TO ASYMMETRIC STABILATOR DEFLECTION)
C
```

```
      DATA DRG3/ -15., .0120,   -14., .0102,   -13., .0085,   -12., .0071,
     *           -10., .0048,    -9., .0038,    -8., .0030,    -7., .0023,
     *            -6., .0016,    -5., .0011,    -4., .0007,    -3., .0003,
     *            -2., .0001,    -1., .0000,     0., .0000,     1., .0000,
     *             2., .0001,     3., .0003,     4., .0007,     5., .0011,
     *             6., .0017,     7., .0023,     8., .0030,     9., .0038,
     *            10., .0048,    11., .0059,    12., .0071,    13., .0085,
     *            14., .0100,    15., .0120,   12*9999.  /
C
C        D.  DCDSTR VS. RUDDER (INCREMENT IN DRAG COEFFICIENT
C            DUE TO RUDDER DEFLECTION)
C
      DATA DRG4/ -15., .0059,   -14., .0054,   -13., .0048,   -12., .0043,
     *           -10., .0032,    -9., .0027,    -8., .0022,    -7., .0017,
     *            -6., .0013,    -5., .0009,    -4., .0006,    -3., .0003,
     *            -2., .0002,    -1., .0001,     0., .0000,     1., .0001,
     *             2., .0001,     4., .0006,     5., .0009,     6., .0013,
     *             7., .0017,     8., .0022,     9., .0027,    10., .0032,
     *            11., .0037,    12., .0043,    13., .0048,    14., .0054,
     *            15., .0059,   14*9999.  /
C
C
C  *  3.  PITCHING MOMENT COEFFICIENT DATA
C
C        A.  CMBAS VS. ACA (BASIC PITCHING MOMENT COEFFICIENT
C            AS A FUNCTION OF ANGLE OF ATTACK)
C
      DATA PTCH1/ -10., 1.13,    -8., 0.80,     -6., 0.50,     -4., 0.31,
     *             -2., 0.16,     0., 0.08,      2.,-0.03,      4.,-0.13,
     *              6.,-0.22,     8.,-0.32,     10.,-0.46,     12.,-0.62,
     *            48*9999. /
C
C        B.  DCMSTE VS. STBLTR (INCREMENT IN PITCHING MOMENT
C            COEFFICIENT DUE TO SYMMETRIC STABILATOR DEFLECTION)
C
      DATA PTCH2/ -15., .90,     -14., .88,     -12., .78,     -10., .68,
     *             -8., .56,      -6., .42,      -4., .27,      -2., .12,
     *              0., .00,       2.,-.10,       4.,-.20,       6.,-.32,
     *              8.,-.45,      10.,-.58,      12.,-.69,      14.,-.78,
     *             15.,-.80,     38*9999./
C
C
C  *  4.  SIDESLIP COEFFICIENT DATA
C
C        A.  CYBAS VS. SIDESL AND ACA (BASIC SIDE FORCE COEFFICIENT
C            AS A FUNCTION OF SIDESLIP AND ANGLE OF ATTACK. THE IN-
C            DEPENDENT VARIABLE IS SIDESLIP, THE PARAMETER IS ANGLE
C            OF ATTACK).
```

```
C
      DATA SIC1/  0.0,   -8.0  ,  -6.0  ,   4.0  ,   8.0  ,   9999.,
     *           -8.0,   0.148,   0.149,   0.140,   0.137,   9999.,
     *           -6.0,   0.111,   0.110,   0.100,   0.077,   9999.,
     *           -4.0,   0.073,   0.072,   0.063,   0.060,   9999.,
     *           -2.0,   0.034,   0.034,   0.025,   0.022,   9999.,
     *            0.0,  -0.002,  -0.003,  -0.013,  -0.020,   9999.,
     *            2.0,  -0.040,  -0.042,  -0.050,  -0.060,   9999.,
     *            4.0,  -0.077,  -0.078,  -0.089,  -0.096,   9999.,
     *            6.0,  -0.114,  -0.116,  -0.127,  -0.129,   9999.,
     *            8.0,  -0.151,  -0.153,  -0.162,  -0.162,   9999./
C
C
C         B.  CRBAS VS. SIDESL AND ACA (BASIC ROLL COEFFICIENT AS A
C             FUNCTION OF SIDESLIP AND PARAMETER ANGLE OF ATTACK)
C
      DATA SIC2/  0.0,   -8.0  ,   2.0  ,   6.0  ,  10.0  ,   9999.,
     *           -8.0,   .0048 ,  .0046 ,  .0040 ,  .0044 ,   9999.,
     *           -6.0,   .0038 ,  .0036 ,  .0028 ,  .0030 ,   9999.,
     *           -4.0,   .0026 ,  .0022 ,  .0015 ,  .0020 ,   9999.,
     *           -2.0,   .0014 ,  .0010 ,  .0004 ,  .0008 ,   9999.,
     *            0.0,   .0004 , -.0004 , -.0006 , -.0002 ,   9999.,
     *            2.0,  -.0005 , -.0014 , -.0020 , -.0014 ,   9999.,
     *            4.0,  -.0016 , -.0026 , -.0032 , -.0025 ,   9999.,
     *            6.0,  -.0030 , -.0038 , -.0042 , -.0036 ,   9999.,
     *            8.0,  -.0044 , -.0052 , -.0057 , -.0048 ,   9999./
C
C
C         C.  CNBAS VS. SIDESL AND ACA (BASIC YAW COEFFICIENT AS A
C             FUNCTION OF SIDESLIP AND PARAMETER ANGLE OF ATTACK)
C
      DATA SIC3/  0.0,   -8.0  ,   0.0  ,   6.0  ,   8.0  ,   9999.,
     *           -8.0,  -.034  , -.037  , -.042  , -.042  ,   9999.,
     *           -6.0,  -.025  , -.028  , -.033  , -.033  ,   9999.,
     *           -4.0,  -.016  , -.019  , -.022  , -.023  ,   9999.,
     *           -2.0,  -.007  , -.009  , -.012  , -.014  ,   9999.,
     *            0.0,   .001  , -.001  , -.004  , -.004  ,   9999.,
     *            2.0,   .010  ,  .009  ,  .007  ,  .007  ,   9999.,
     *            4.0,   .019  ,  .018  ,  .016  ,  .016  ,   9999.,
     *            6.0,   .028  ,  .026  ,  .024  ,  .024  ,   9999.,
     *            8.0,   .036  ,  .034  ,  .032  ,  .032  ,   9999./
C
C
C  *  5.  DIRECTIONAL CONTROL COEFFICIENT DATA
C
C         A.  DCYSTR VS. RUDDER AND ACA (INCREMENT IN SIDE FORCE
C             COEFFICIENT DUE TO RUDDER DEFLECTION AND PARAMETER
C             ANGLE OF ATTACK)
```

```
C
      DATA CREC1/  0.0,    -8.0    ,   2.0    ,   6.0    ,  8.0    ,     9999.,
     *            -15.0,   -.090   , -.097    , -.095    ,-.102    ,     9999.,
     *            -10.0,   -.060   , -.062    , -.062    ,-.069    ,     9999.,
     *             -5.0,   -.032   , -.036    , -.037    ,-.042    ,     9999.,
     *              0.0,    .004   , -.002    , -.003    ,-.003    ,     9999.,
     *              5.0,    .040   ,  .042    ,  .038    , .004    ,     9999.,
     *             10.0,    .070   ,  .066    ,  .062    , .066    ,     9999.,
     *             15.0,    .100   ,  .098    ,  .092    , .096    ,     9999.,
     * 12*9999./
C
C
C         B.  DCNSTR VS. RUDDER AND AOA (INCREMENT IN YAWING MOMENT
C             COEFFICIENT DUE TO RUDDER DEFLECTION AND PARAMETER
C             ANGLE OF ATTACK)
C
      DATA CREC2/  0.0,    -8.0    ,   2.0    ,   6.0    ,  8.0    ,     9999.,
     *            -15.0,    .070   ,  .068    ,  .062    , .065    ,     9999.,
     *            -10.0,    .050   ,  .049    ,  .048    , .048    ,     9999.,
     *             -5.0,    .026   ,  .024    ,  .022    , .018    ,     9999.,
     *              0.0,    .000   ,  .000    ,  .000    , .000    ,     9999.,
     *              5.0,   -.022   , -.020    , -.018    ,-.014    ,     9999.,
     *             10.0,   -.052   , -.048    , -.047    ,-.046    ,     9999.,
     *             15.0,   -.080   , -.079    , -.074    ,-.075    ,     9999.,
     * 12*9999./
C
C
C         C.  DCRSTR VS. RUDDER AND AOA (INCREMENT IN ROLLING MOMENT
C             COEFFICIENT DUE TO RUDDER DEFLECTION AND PARAMETER
C             ANGLE OF ATTACK)
C
      DATA CREC3/  0.0,    -8.0    ,   2.0    ,   4.0    ,  6.0    ,   8.0    ,
     *            -15.0,   -.0050  , -.0038   , -.0032   ,-.0022   ,-.0002   ,
     *            -10.0,   -.0034  , -.0029   , -.0026   ,-.0020   ,-.0004   ,
     *             -5.0,   -.0018  , -.0018   , -.0018   ,-.0014   ,-.0008   ,
     *              0.0,    .0004  ,  .0000   , -.0003   ,-.0008   ,-.0013   ,
     *              5.0,    .0024  ,  .0012   ,  .0004   ,-.0005   ,-.0014   ,
     *             10.0,    .0048  ,  .0028   ,  .0016   , .0000   ,-.0016   ,
     *             15.0,    .0070  ,  .0053   ,  .0040   ,-.0012   ,-.0025   ,
     * 12*9999./
C
C
C  *  6.  LATERAL CONTROL COEFFICIENT DATA
C
C         A.  DCYSTA VS. AILRON AND ACA (INCREMENT IN SIDE FORCE
C             COEFFICIENT DUE TO ASYMETRIC STABILATOR DEFLECTION AND
C             PARAMETER ANGLE OF ATTACK)
C
```

```
      DATA LTRL1/  0.0,    -8.0   ,  -1.0   ,   4.0   , 10.0   ,     9999.,
     *          -15.0,    .015   ,  .019   ,  .017   ,  .006   ,     9999.,
     *          -10.0,    .013   ,  .014   ,  .010   ,-.002   ,     9999.,
     *           -5.0,    .010   ,  .009   ,  .002   ,-.010   ,     9999.,
     *            0.0,    .007   ,  .002   , -.007   ,-.018   ,     9999.,
     *            5.0,    .002   , -.003   , -.014   ,-.026   ,     9999.,
     *           10.0,   -.005   , -.009   , -.020   ,-.033   ,     9999.,
     *           15.0,   -.010   , -.014   , -.025   ,-.042   ,     9999.,
     * 12*9999./
C
C
C          B.   DCNSTA VS. AILRON AND AOA (INCREMENT IN YAWING MOMENT
C               COEFFICIENT DUE TO ASYMETRIC STABILATOR DEFLECTION AND
C               PARAMETER ANGLE OF ATTACK)
C
      DATA LTRL2/  0.0,    -8.0   ,  -4.0   ,   0.0   ,  4.0   ,   8.0    ,
     *          -15.0,   .0075   , .0050   ,  .00    ,-.0075  ,-.0120   ,
     *          -10.0,   .0040   , .0030   ,  .00    ,-.0050  ,-.0085   ,
     *           -5.0,   .0015   , .0020   ,  .00    ,-.0025  ,-.0060   ,
     *            0.0,  -.0020   ,-.0010   ,  .00    ,-.0008  ,-.0020   ,
     *            5.0,  -.0070   ,-.0030   ,  .00    , .0025  , .0050   ,
     *           10.0,  -.0120   ,-.0045   ,  .00    , .0040  , .0120   ,
     *           15.0,  -.0175   ,-.0070   ,  .00    , .0070  , .0175   ,
     * 12*9999./
C
C          C.   DCRSTA VS. AILRON AND AOA (INCREMENT IN ROLLING MOMENT
C               COEFFICIENT DUE TO ASYMETRIC STABILATOR DEFLECTION AND
C               PARAMETER ANGLE OF ATTACK)
C
      DATA LTRL3/  0.0,    -8.0   ,  -4.0   ,   0.0   ,  4.0   ,   8.0    ,
     *          -15.0,   .0087   , .0096   , .0100   , .0092  , .0084   ,
     *          -10.0,   .0060   , .0066   , .0068   , .0064  , .0058   ,
     *           -5.0,   .0030   , .0033   , .0034   , .0036  , .0030   ,
     *            0.0,   .0004   , .0004   , .0004   , .0030  , .0030   ,
     *            5.0,  -.0012   ,-.0024   ,-.0030   ,-.0030  ,-.0020   ,
     *           10.0,  -.0030   ,-.0058   ,-.0072   ,-.0046  ,-.0047   ,
     *           15.0,  -.0050   ,-.0080   ,-.0092   ,-.0088  ,-.0067   ,
     * 12*9999./
C
C
C
C ** 2. DYNAMIC STABILITY DERIVATIVES
C
      DATA  CLADT        ,CDADT        ,CMADT
     *      /2.00        ,0.10         ,-10.0   /,
     *       CLQ         ,CDQ          ,CMQ
     *      /5.0         ,0.1          ,-15.0   /,
     *       CRBDT       ,CYR          ,CNBDT
```

```
     *       /0.0          ,0.4          ,0.15     /,
     *        CRR          ,CYP          ,CNR
     *       /0.2          ,-0.1         ,-0.2     /,
     *        CRP          ,CYBDT        ,CNP
     *       /-0.4         ,-0.1         ,-0.01    /
C
C
C
C      *****************************************
C      COMMON BLOCK /C/: CONTROL SYSTEM PARAMETERS
C      *****************************************
C
C *** CONTROL SYSTEM PARAMETERS
C
       DATA          KPTCHR        ,KROLLR       ,KYAWRT       ,KBANK
     *              /0.28          ,0.10         ,0.40         ,10.8    /,
     *               KGAMMA        ,KALT         ,CGARML       ,CGARMN
     *              /1.0           ,0.3          ,0.0          ,0.0     /,
     *               RRTLIM        ,KNY          ,KNZ
     *              /75.0          ,0.35         ,0.05/
C
C
C      *****************************************
C      COMMON BLOCK /E/: INITIAL CONDITIONS
C      *****************************************
C
C
C *** MISSILE INITIAL CONDITIONS
C
       DATA          XM1           ,YM1          ,ALTUD1       ,U1
     *              /0.0           ,0.0          ,50.0         ,838.1/
C
       DATA          PITCH1        ,BANK1        ,HEDNG1       ,SIDES1
     *              /3.00          ,0.0          ,0.0          ,0.0     /
C
       DATA          AUA1          ,PTCHR1       ,ROLLR1       ,YAWR1
     *              /3.00          ,0.0          ,0.0          ,0.0     /
C
C
C *** TARGET INITIAL CONDITIONS
C
       DATA          XT1           ,YT1          ,HT1          ,TSPEED
     *              /24000.0       ,0.0          ,10.0         ,35.0    /
C
C
C
C      *****************************************
C      COMMON BLOCK /F/: GUIDANCE PARAMETERS
```

```
C       ****************************************************
C
C
C *** PROPORTIONAL NAVIGATION & NAV FILTER CONSTANTS (AZIMUTH & ELEV.)
C
      DATA         LAMDAZ        ,LAMDEL       ,KNFAZ       ,KNFEL
     *             /3.15         ,3.85         ,0.8         ,0.50 /
C
C
C *** MISSION PHASE FLAGS & DECISION PARAMETERS
C
      DATA         PH1           ,PH2          ,PH3         ,PH4
     *             /0            ,0            ,0           ,0     /
C
      DATA         OFFSET        ,ALTATT       ,SGDZPU      ,MISDST
     *             /10.0         ,200.0        ,1.12        ,0.0   /
C
C
C
C       ****************************************************
C       COMMON BLOCK /H/: ECM/GLINT PARAMETERS
C       ****************************************************
C
      DATA         FREQ          ,SHIFTY       ,SHIFTH      ,BRNTHR
     *             /0.00         ,-75.0        ,10.0        ,250.0 /
C
      END
C
C
C                                                                   9-07-84
C *************************************************************************
      SUBROUTINE MISSN
C *************************************************************************
C     MAKES MISSION PHASE DECISIONS AND INVOKES THE DIFFERENT MODES
C     OF GUIDANCE AS REQUIRED. DELIVERS NZC AND PHIC TO THE AUTOPILOT
C     CONTROL LOOPS. NYC IS ASSUMED TO ALWAYS BE ZERO EXCEPT DURING
C     ATTACK. NZC IS LIMITED TO +4.0 AND -2.0 G'S.
C     *********************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NCLT,PCOUNT,NFAZE
C
C
C     ****************************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     ****************************************************
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPUT  ,NCLT  ,NPTS  ,CPA   ,PCOUNT,
     *           G      ,T           ,RHO   ,PI    ,PII   ,MASS  ,WT    ,S     ,
```

```
     *             IXX    ,IYY    ,IZZ  ,IXZ  ,IA    ,IB    ,IC    ,QS    ,
     *             ID     ,IE     ,IF   ,IG   ,IH    ,II    ,IJ    ,IK    ,
     *             CHORD2,CHORD ,SPAN2,SPAN ,NFAZE
C
C
C     ****************************************
C     COMMON BLOCK /C/: CONTROL SYSTEM PARAMETERS
C     ****************************************
C
      COMMON /C/  KPTCHR        ,KROLLR        ,KYAWRT        ,KBANK       ,
     *            KGAMMA        ,KALT          ,CGARML        ,CGARMN      ,
     *            RRTLIM        ,PLIM          ,KNY           ,KNZ         ,
     *            AILRON        ,STBLTR        ,RUDDER        ,
     *            BSERO         ,NZSERO        ,NYSERO
C
C     ****************************************
C     COMMON BLOCK /D/: MISSILE FLIGHT DYNAMICS PARAMETERS
C     ****************************************
C
      COMMON /D/  ALFA          ,BETA          ,VT            ,HMDOT       ,
     *            U             ,V             ,W             ,
     *            PHI           ,GAMMA         ,THETA         ,SY          ,
     *            CD            ,CY            ,CL            ,CR          ,
     *            CM            ,CN            ,P             ,Q           ,
     *            R             ,ALFADT        ,BETADT        ,PDOT        ,
     *            QDOT          ,RDOT          ,NZ            ,ALTUDE      ,
     *            XM            ,YM            ,XMDOT         ,YMDOT
C
C     ****************************************
C     COMMON BLOCK /F/: GUIDANCE PARAMETERS
C     ****************************************
C
      COMMON /F/  PH1           ,PH2           ,PH3           ,PH4         ,
     *            OFFSET        ,ALTATT        ,SGDZPU        ,MISDST      ,
     *            LAMDAZ        ,LAMDEL        ,KNFAZ         ,KNFEL       ,
     *            NZC           , PHIC         ,GAMMAC        ,PCLIM       ,
     *            PC            ,QC            ,RC            ,RANGE       ,
     *            SIGAZ         ,SIGEL         ,SIGDAF        ,SIGDEF      ,
     *            SYT           ,THETAT        ,XT            ,YT          ,
     *            FT            ,NYC           ,PUPRNG
C
C     ****************************************
C     COMMON BLOCK /G/: OUTPUT PARAMETERS
C     ****************************************
C
      COMMON /G/  AOA           ,SIDESL        ,BANK          ,FLTPHC      ,
     *            BANKC         ,PITCH         ,ROLLRT        ,ROLRTC      ,
     *            PTCHRT        ,YAWRT         ,HEADNG        ,FLTPTH      ,
```

```
     *              HEADT          ,ELEVT        ,DSIGAZ       ,DSIGEL      ,
     *              DSGDAZ         ,DSGDEL       ,ERFEK        ,ERFRR       ,
     *              ERFAZ          ,ERFEL
C
C
C***********************************************************************
C
C      MISSION PHASE LOGIC AND GUIDANCE COMMANDS
C
    1 IF(PH4.EQ.1) GO TO 40
      IF(PH3.EQ.1) GO TO 30
      IF(PH2.EQ.1) GO TO 20
      IF(PH1.EQ.1) GO TO 10
C
C *** INGRESS FROM INITIAL CONDITION TO OFFSET MANEUVER
C
         IF(RANGE.LT.18000.0) GO TO 9
C
C           ALTITUDE HOLD
C
C
            ALTC   = 50.0
            ALTUDF = ALTUDE
            GAMMAC = KALT*(ALTC-ALTUDF)/VT
            GAMMAF = GAMMA
            AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C           PROPORTIONAL NAVIGATION IN AZIMUTH
C
            AYC    = LAMDAZ*VT*SIGDAF/G
            PHIC   = ATAN2(AYC,AZC)
            AZC    = AZC*COS(PHI)+AYC*SIN(PHI)
            GO TO 100
    9       PH1    = 1
C
C *** OFFSET TURN (60 DEG BANK) TO OFFSET HEADING
C
   10    ABSDSZ = ABS(DSIGAZ)
         IF(ABSDSZ.GT.OFFSET) GO TO 19
C
C           ALTITUDE HOLD
C
            ALTC   = 50.0
            ALTUDF = ALTUDE
            GAMMAC = KALT*(ALTC-ALTUDF)/VT
            GAMMAF = GAMMA
            AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
```

```
C              BANK ANGLE HOLD (60 DEG)
C
               AYC    = 0.0
               PHIC   = 60.0/PII
               AZC    = AZC/COS(PHI)
               GO TO 100
   19          PH2    = 1
C
C  ***     COURSE HOLD ON OFFSET HEADING TO POPUP
C
   20      ABDSDZ = ABS(DSGDAZ)
           IF(ABDSDZ.GT.SGDZPU) GO TO 29
C
C              ALTITUDE HOLD
C
               ALTC   = 50.0
               ALTUDF = ALTUDE
               GAMMAC = KALT*(ALTC-ALTUDF)/VT
               GAMMAF = GAMMA
               AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C              BANK ANGLE HOLD (0 DEG)
C
               AYC    = 0.0
               PHIC   = 0.0
               AZC    = AZC/COS(PHI)
               GO TO 100
   29          PH3    = 1
C
C  ***     PULLUP TO ATTACK ALTITUDE
C          PROPORTIONAL NAVIGATION IN AZIMUTH
C
   30      IF(ALTUDE.GT.ALTATT) GO TO 39
C
C              VERTICAL FLIGHT PATH ANGLE HOLD (8.5 DEG)
C
               ALTC   = 0.0
               GAMMAC = 8.5/PII
               GAMMAF = GAMMA
               AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C              PROPORTIONAL NAVIGATION IN AZIMUTH
C
               AYC    = LAMDAZ*VT*SIGDAF/G
               AZC    = AZC*COS(PHI)+AYC*SIN(PHI)
               PHIC   = ATAN2(AYC,AZC)
               GO TO 100
   39          PH4    = 1
```

```
C
C ***    ATTACK
C        PROPORTIONAL NAVIGATION IN AZIMUTH AND ELEVATION
C
   40        ALTC   = 0.0
             GAMMAC = 0.0
             GAMMAF = GAMMA
             AYC    = LAMDAZ*VT*SIGDAF/G
             AZC    = LAMDEL*VT*SIGDEF/G+COS(GAMMAF)
             NZC    = AZC*COS(PHI)+AYC*SIN(PHI)
             NYC    = 0.0
C
C            BANK ANGLE COMMAND ROUTINE INSURES ROLL
C            IN SHORTEST DIRECTION
C
             PHIC   = ATAN2(AYC,AZC)
             DELPHI = PHIC-PHI
             DPHIAB = ABS(DELPHI)
             IF(DPHIAB.LT.PI) GO TO 100
             IF(PHIC.GE.0.0) GO TO 90
             PHIC   = PHIC+2.0*PI
             GO TO 100
   90        PHIC   = PHIC-2.0*PI
  100 CONTINUE
C ********************************************************************************
C
C *** NZ COMMAND LIMITED TO -2 & +4 G'S
C
      NZC  = LIMIT(-2.0,4.0,NZC)
C
      RETURN
      END
C
C                                                                        9-07-84
C ***************************************************************************
      SUBROUTINE APILOT
C ***************************************************************************
C     LIMIT, REALFL
C   *************************************************************************
C     MODELS THE INNER LOOP AUTOPILOT AND CONTROL MIXER. CALCULATES
C     REQUIRED ELEVATOR, AILERON & RUDDER REQUIRED, MIXES THESE TO GET
C     THE FIN STABILATOR COMMANDS, APPLIES THE FIN LIMITS OF +-15 DEG.
C     AND ADJUSTS THE THREE CONTROL OUTPUTS TO ACCOUNT FOR THE LIMITS.
C     **********************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOUT,PCOUNT,NFAZE
C
```

```
C
C     ******************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     ******************************************
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPDT ,NCUT ,NPTS ,CPA  ,PCOUNT,
     *           G      ,T     ,RHO   ,PI   ,PII  ,MASS ,WT   ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ  ,IA   ,IB   ,IC   ,GS    ,
     *           ID     ,IE    ,IF    ,IG   ,IH   ,II   ,IJ   ,IK    ,
     *           CHORD2,CHORD ,SPAN2 ,SPAN ,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /C/: CONTROL SYSTEM PARAMETERS
C     ******************************************
C
      COMMON /C/ KPTCHR        ,KRCLLR       ,KYAWRT       ,KBANK        ,
     *           KGAMMA        ,KALT         ,CGARML       ,CGARMN       ,
     *           RRTLIM        ,PLIM         ,KNY          ,KNZ          ,
     *           AILRON        ,STBLTR       ,RUDDER       ,
     *           BSERD         ,NZSERD       ,NYSERD
C
C
C     ******************************************
C     COMMON BLOCK /D/: MISSILE FLIGHT DYNAMICS PARAMETERS
C     ******************************************
C
      COMMON /D/ ALFA          ,BETA         ,VT           ,HMDOT        ,
     *           U             ,V            ,W            ,
     *           PHI           ,GAMMA        ,THETA        ,SY           ,
     *           CD            ,CY           ,CL           ,CR           ,
     *           CM            ,CN           ,P            ,Q            ,
     *           R             ,ALFADT       ,BETADT       ,PDOT         ,
     *           UDOT          ,RDOT         ,NZ           ,ALTUDE       ,
     *           XM            ,YM           ,XMDCT        ,YMDCT
C
C
C     ******************************************
C     COMMON BLOCK /F/: GUIDANCE PARAMETERS
C     ******************************************
C
      COMMON /F/ PH1           ,PH2          ,PH3          ,PH4          ,
     *           OFFSET        ,ALTATT       ,SGDZPU       ,MISDST       ,
     *           LAMDAZ        ,LAMDEL       ,KNFAZ        ,KNFEL        ,
     *           NZC           , PHIC        ,GAMMAC       ,PCLIM        ,
     *           PC            ,QC           ,RC           ,RANGE        ,
     *           SIGAZ         ,SIGEL        ,SIGDAF       ,SIGDEF       ,
     *           SYT           ,THETAT       ,XT           ,YT           ,
```

```
     *              FT              ,NYC          ,PUPRNG
C
C EXECUTABLE STMTS ***************************************************
C
C      INNER LCCP AUTOPILCT
C
C         NORMAL ACCELERATIUN COMMAND SYSTEM
C
          NZCLIM = LIMIT(-2.0,4.0,NZC)
          NZZ    = NZ+CGARMN*QDOT/G
          ENZ    = NZCLIM-NZZ
          ENZKNZ = ENZ*KNZ
          QC     = QC + ENZKNZ*DT
          NZSERI = QC-KPTCHR*Q
          NZSERO = REALPL(NZSERO,0.025,NZSERI,DT)
C
          ELE    = -PII*NZSERO
C
C
C         BANK ANGLE COMMAND SYSTEM
C
          EPHI   = PHIC-PHI
          PC     = KBANK*EPHI
          PCLIM  = LIMIT(-PLIM,PLIM,PC)
          EP     = PCLIM-P
          BSERI  = EP*KROLLR
          BSERO  = REALPL(BSERO,0.025,BSERI,DT)
C
          AIL    = -PII*BSERO
C
C
C         TURN COORDINATOR
C
          IF(PH4.EQ.1) GO TO 30
             NYC    = 0.0
   30     CONTINUE
          NYY    = NY+CGARML*RDOT/G
          ENY    = NYC-NYY
          RC     = RC+ENY*KNY*DT
          NYSERI = RC-KYAWRT*R
          NYSERO = REALPL(NYSERO,0.025,NYSERI,DT)
C
          RUD    = -PII*NYSERO
C      CONTROLS MIXER AND LIMITS
C
       LSTAB1 = ELE-AIL
       RSTAB1 = ELE+AIL
       LSTAB  = LIMIT(-15.0,15.0,LSTAB1)
```

```
      RSTAB   = LIMIT(-15.0,15.0,RSTAB1)
C
      STBLTR  = (RSTAB+LSTAB)/2.0
      AILRON  = (RSTAB-LSTAB)/2.0
      RUDDER  = LIMIT(-15.0,15.0,RUD)
C
C
      RETURN
      END
C
C                                                                   8-01-84
C **********************************************************************
      FUNCTION LIMIT(LOW,HI,CONTRL)
C **********************************************************************
C     MODELS THE PHYSICAL LIMITS IMPOSED UPON THE CONTROLS AND OTHER
C     PARAMETERS.
C     *****************************************************************
C
      IMPLICIT REAL(A-Z)
C     *******************************************************************
C
      IF (CONTRL.GT.LOW) GO TO  50
          LIMIT = LOW
          RETURN
   50 IF (CONTRL.LT.HI ) GO TO 100
          LIMIT = HI
          RETURN
  100 CONTINUE
          LIMIT = CONTRL
          RETURN
      END
C
C                                                                   8-04-84
C **********************************************************************
      FUNCTION REALPL(PV,T,INPUT,DT)
C **********************************************************************
C     MODELS A FIRST ORDER LAG (Y/X = 1/TS+1).
C     *****************************************************************
C
      IMPLICIT REAL(A-Z)
C
      PVDOT   = (1/T)*(INPUT-PV)
      REALPL  = PV + PVDOT*DT
      RETURN
      END
C                                                                   9-07-84
C **********************************************************************
      SUBROUTINE AERO
```

```
C ******************************************************************
C     TABLE1,TABLE2,SUPRES
C    ***************************************************************
C     USES TABLE LOOKUP ROUTINES (TABLE1,2) TO CONSTRUCT THE AERO-
C     DYNAMIC COEFFICIENTS FOR THE MISSILE GIVEN THE CONTROL
C     INPUTS, ANGLE OF ATTACK, SIDESLIP AND ANGULAR RATES P,Q,R,ADT,BDT.
C     THEN CALCULATES MOMENTS, FORCES, RATES, ANGLES AND POSITIONS OF
C     THE MISSILE IN SPACE. ALL VARIABLES RETURNED ARE RETURNED WITHIN
C     THE COMMON BLOCKS. OUT-OF-RANGE WARNINGS ARE SUPPRESSED AFTER
C     20 CONSECUTIVE CALLS OUT OF RANGE.
C    ***************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOLT,PCOUNT,NFAZE
C
C
C     *******************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     *******************************************
C
      COMMON /A/ TIME   ,FINTIM,DT    ,UPDT ,NOLT ,NPTS ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI   ,PII  ,MASS ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ  ,IA   ,IB   ,IC    ,CS    ,
     *           ID     ,IE    ,IF    ,IG   ,IH   ,II   ,IJ    ,IK    ,
     *           CHORD2,CHORD  ,SPAN2 ,SPAN ,NFAZE
C
C
C     *******************************************
C     COMMON BLOCK /B/: AERODYNAMIC COEFFICIENT TABLES
C     *******************************************
C
      COMMON /B/ LFT1(2,36)  ,LFT2(2,36)  ,DRG1(2,36)  ,DRG2(2,36)  ,
     *           DRG3(2,36)  ,DRG4(2,36)  ,PTCH1(2,36) ,PTCH2(2,36) ,
     *           SID1(6,10)  ,SID2(6,10)  ,SID3(6,10)  ,DREC1(6,10) ,
     *           DREC2(6,10) ,DREC3(6,10) ,LTRL1(6,10) ,LTRL2(6,10) ,
     *           LTRL3(6,10) ,CLADT       ,CDADT       ,CMADT       ,
     *           CLQ         ,CDQ         ,CMQ         ,CRBDT       ,
     *           CYR         ,CNBDT       ,CRR         ,CYP         ,
     *           CNR         ,CRP         ,CYBDT       ,CNP
C
C
C     *******************************************
C     COMMON BLOCK /C/: CONTROL SYSTEM PARAMETERS
C     *******************************************
C
      COMMON /C/ KPTCHR      ,KROLLR      ,KYAWRT      ,KBANK       ,
     *           KGAMMA      ,KALT        ,LGARML      ,LGARMN      ,
     *           RRTLIM      ,PLIM        ,KNY         ,KNZ         ,
```

```
     *          AILRON        ,STBLTR       ,RUDCER       ,
     *          BSERO         ,NZSERO       ,NYSERO
C
C
C    ************************************************
C    COMMON BLOCK /D/: MISSILE FLIGHT DYNAMICS PARAMETERS
C    ************************************************
C
     COMMON /C/ ALFA          ,BETA         ,VT           ,HMDOT         ,
     *          U             ,V            ,W            ,
     *          PHI           ,GAMMA        ,THETA        ,SY            ,
     *          CD            ,CY           ,CL           ,CR            ,
     *          CM            ,CN           ,P            ,Q             ,
     *          R             ,ALFADT       ,BETADT       ,PDOT          ,
     *          UDOT          ,RDCT         ,NZ           ,ALTUDE        ,
     *          XM            ,YM           ,XMDCT        ,YMDOT
C
C
C    ******************************************************************
     SIDESL = PII*BETA
     AOA = PII*ALFA
C
     CLEAS  = TABLE1(LFT1,AOA)
     DCLSTE = TABLE1(LFT2,STBLTR)
     CDBAS  = TABLE1(DRG1,CLBAS)
     DCDSTE = TABLE1(DRG2,STBLTR)
     DCDSTA = TABLE1(DRG3,AILRON)
     DCDSTR = TABLE1(DRG4,RUDDER)
     CMBAS  = TABLE1(PTCH1,AOA)
     DCMSTE = TABLE1(PTCH2,STBLTR)
     CYBAS  = TABLE2(SID1,AOA,SIDESL)
     CRBAS  = TABLE2(SID2,AOA,SIDESL)
     CNBAS  = TABLE2(SID3,AOA,SIDESL)
     DCYSTR = TABLE2(DREC1,AOA,RUDDER)
     DCNSTR = TABLE2(DREC2,AOA,RUDDER)
     DCRSTR = TABLE2(DREC3,AOA,RUDDER)
     DCYSTA = TABLE2(LTRL1,AOA,AILRON)
     DCNSTA = TABLE2(LTRL2,AOA,AILRON)
     DCRSTA = TABLE2(LTRL3,AOA,AILRON)
C
C    AERODYNAMIC COEFFICIENTS
C
     CL  =  CLEAS+DCLSTE+CHORD2*(CLADT*ALFADT+CLQ*Q)/VT
     CD  =  CDEAS+DCDSTE+DCDSTA+DCDSTR+CHORD2*(CDADT*ALFADT+CDQ*Q)/VT
     CY  =  CYEAS+DCYSTR+DCYSTA+SPAN2*(CYR*R+CYP*P+CYBDT*BETADT)/VT
     CM  =  CMEAS+DCMSTE+CHORD2*(CMADT*ALFADT+CMQ*Q)/VT
     CN  =  CNEAS+DCNSTR+DCNSTA+SPAN2*(CNBDT*BETADT+CNR*R+CNP*P)/VT
     CR  =  CREAS+DCRSTA+DCRSTR+SPAN2*(CRBDT*BETADT+CRR*R+CRP*P)/VT
```

```
C
C
C
C     AERODYNAMIC FORCES AND MOMENTS
C
      L  = CL*QS
      D  = CD*QS
      LA = SPAN*CR*QS
      MA = CHORD*CM*QS
      NA = SPAN*CN*QS
      X  = L*SIN(ALFA)-D*COS(ALFA)
      Y  = CY*QS
      Z  = -L*COS(ALFA)-D*SIN(ALFA)
C
C     NORMAL & LATERAL ACCELERATIONS
C
      NZ = -Z/(MASS*G)
      NY = Y/(MASS*G)
C
C *** COMMENCE INTEGRATION OF EQUATIONS OF MOTION
C
C     EULER ANGLES
C
      PHIDOT = P+TAN(THETA)*(Q*SIN(PHI)+R*COS(PHI))
      THETAD = Q*COS(PHI)-R*SIN(PHI)
      SYDOT  = (Q*SIN(PHI)+R*COS(PHI))/COS(THETA)
      PHI    = PHI + PHIDOT*DT
      THETA  = THETA + THETAD*DT
      SY     = SY + SYDOT*DT
C
C     LINEAR ACCELERATIONS AND VELOCITIES
C
      UDOT = -G*SIN(THETA)+V*R-W*Q+X/MASS+T/MASS
      VDOT = G*SIN(PHI)*COS(THETA)-U*R+W*P+Y/MASS
      WDOT = G*COS(PHI)*COS(THETA)+U*Q-V*P+Z/MASS
      U = U + UDOT*DT
      V = V + VDOT*DT
      W = W + WDOT*DT
      VT = SQRT(U**2+V**2+W**2)
      VTDOT = SQRT(UDOT**2+VDOT**2+WDOT**2)
C
C     ANGULAR ACCELERATIONS AND VELOCITIES
C
      PDOT = IB*LA+IC*NA-ID*P*Q-IE*R*Q
      QDOT = IF*MA-IG*P*R-IH*(P**2-R**2)
      RDOT = IC*LA+II*NA-IJ*P*Q-IK*R*Q
      P = P + PDOT*DT
      Q = Q + QDOT*DT
```

```
      R = R + RDOT*DT
C
C     ANGLES OF ATTACK, SIDESLIP, AND FLIGHT PATH
C
      ALFADT = (ATAN(W/U)-ALFA)/DT
      BETADT = (ASIN(V/VT)-BETA)/DT
      ALFA   = ATAN(W/U)
      BETA   = ASIN(V/VT)
      GAMMA  = ASIN(HMDOT/VT)
C
C     MISSILE POSITION IN INERTIAL SPACE
C
      XMDOT  = U*COS(SY)*COS(THETA)+V*(COS(SY)*SIN(THETA)*SIN(PHI)
     *         -SIN(SY)*COS(PHI))+W*(COS(SY)*SIN(THETA)*COS(PHI)+
     *         SIN(SY)*SIN(PHI))
      YMDOT  = U*SIN(SY)*COS(THETA)+V*(SIN(SY)*SIN(THETA)*SIN(PHI)
     *         +COS(SY)*COS(PHI))+W*(SIN(SY)*SIN(THETA)*COS(PHI)-
     *         COS(SY)*SIN(PHI))
      HMDOT  = U*SIN(THETA)-V*COS(THETA)*SIN(PHI)-W*COS(THETA)*COS(PHI)
      HORDOT= SQRT(XMDOT**2+YMDOT**2)
      XM = XM + XMDOT*DT
      YM = YM + YMDOT*DT
      ALTUDE = ALTUDE + HMDOT*DT
C
C
      RETURN
      END
C
C
C                                                              9-07-84
      ************************************************************************
      FUNCTION    TABLE1 (ARRAY,IP)
      ************************************************************************
C     TABLE LOOKUP WITH LINEAR INTERPOLATION FOR A FUNCTION
C     OF ONE VARIABLE, Y=F(X). MAXIMUM NUMBER OF POINTS IS
C     LIMITED TO 30 BY THE DIMENSION STATEMENT. NOTE: THE
C     FIRST ARRAY ENTRY (ARRAY(1,1)) CANNOT BE 9999.0 AND ALL
C     UNUSED ARRAY ELEMENTS MUST BE EXACTLY 9999. THE INDEPENDENT
C     VARIABLE DATA IS STORED IN ROW 1, THE DEPENDENT IN ROW 2.
C     WARNINGS ARE SUPPRESSED AFTER 20 CONSECUTIVE CALLS (ABOUT 5 DT'S)
C     ************************************************************************
C
      REAL ARRAY(2,30),IP,C,TABLE1
      INTEGER  I,J,K,N,SUPRES
      DATA J,K,N /3*0/
C
      K  = K+1
      IF (.NOT.IP.LT.ARRAY(1,1)) GO TO 10
          TABLE1=ARRAY(2,1)
```

```
C             IF(SUPRES(J,K,N).EQ.1)  WRITE(6,101) IP,TABLE1
              RETURN
C         END IF
C
C
   10 DO 90  I=1,30
              IF(.NOT.ARRAY(1,I).EQ.9999.0) GO TO 20
                  TABLE1=ARRAY(2,I-1)
C                 IF(SUPRES(J,K,N).EQ.1)  WRITE(6,102) IP,TABLE1
                  RETURN
   20         IF(.NOT.IP.GT.ARRAY(1,I)) GO TO 30
                  GO TO 90
   30         IF(.NOT.IP.EQ.ARRAY(1,I)) GO TO 40
                  TABLE1 = ARRAY(2,I)
                  RETURN
   40         IF(.NOT.IP.LT.ARRAY(1,I)) GO TO 90
                  C= (IP-ARRAY(1,I-1))/(ARRAY(1,I)-ARRAY(1,I-1))
                  TABLE1= ARRAY(2,I-1) +C*(ARRAY(2,I)-ARRAY(2,I-1))

                  RETURN
C             END IF
   90 CONTINUE
C     WRITE(6,103)
      RETURN
C
C     FORMAT STATEMENTS FOR SUBROUTINE TABLE1*****************************
C
  101 FORMAT('0','SUBROUTINE TABLE1: INPUT BELOW INDEPENDENT VARIABLE DA
     *TA.'/'          INPUT (IP)                  = ',F10.2,
     *    /'          USED LOWEST DATA AVAILABLE  = ',F10.2)
  102 FORMAT('0','SUBROUTINE TABLE1: INPUT ABOVE INDEPENDENT VARIABLE DA
     *TA.'/'          INPUT (IP)                  = ',F10.2,
     *    /'          USED HIGHEST DATA AVAILABLE = ',F10.2)
  103 FORMAT('0','SUBROUTINE TABLE1: ERROR. SUBROUTINE DID NOT END ')
      END
C
C                                                              9-07-84
C     *********************************************************************
      FUNCTION TABLE2 (A,IP,IV)
C     *********************************************************************
C     TABLE LOOKUP WITH LINEAR INTERPOLATION FOR A FUNCTION
C     OF TWO VARIABLES, Z=F(X,Y). MAXIMUM NUMBER OF INPUT PARAMETERS IS
C     LIMITED TO 5, INDEPENDENT VARIABLE DATA POINTS TO 9 BY THE
C     DIMENSION OF THE ARRAY. A(1,1) IS NOT USED. THE PARAMETER VALUES
C     ARE STORED IN COL 1, INDEP. VAR. VALUES IN ROW 1. THE COEFFICIENT
C     DATA ARE STORED IN THE GRID CREATED BY ROW AND COL 1.
C     TABLE2 WARNINGS ARE SUPPRESSED AFTER 5 CONSECUTIVE OCCURRENCES.
C     *********************************************************************
```

```
C
      REAL A(6,10),IP,IV,CP,CV,LFT,RGT,TABLE2
      INTEGER  I,LI,UI,J,K,N,SUPRES
      DATA J,K,N /3*0/
C
C
      K   = K+1
      IF(.NOT.IP.LT.A(2,1)) GO TO 10
            CF = 0.
            LI = 2
            UI = 2
C           IF(SUPRES(J,K,N).EQ.1)  WRITE(6,1001) IP,A(2,1)
            GO TO 55
C
 10         DO 50 I=2,6
                  IF(.NOT.A(I,1).EQ.9999.0) GO TO 20
C                       IF(SUPRES(J,K,N).EQ.1)  WRITE(6,1002) IP,A(I-1,1)
                        CP = 0.
                        LI = I-1
                        UI = I-1
                        GO TO 55
C
 20               IF(IP-A(I,1)) 30,40,50
C                 IP < A(I,1)
 30                     CP = (IP-A(I-1,1))/(A(I,1)-A(I-1,1))
                        LI = I-1
                        UI = I
                        GO TO 55
C                 IF =  A(I,1)
 40                     CP = 0
                        LI = I
                        UI = I
                        GO TO 55
C                 END IF
 50         CONTINUE
C     END IF
C
 55   IF(.NOT.IV.LT.A(1,2)) GO TO 60
C           IF(SUPRES(J,K,N).EQ.1)  WRITE(6,1003) IV,A(1,2)
            TAELE2=A(LI,2)+CP*(A(UI,2)-A(LI,2))
            RETURN
C
 60         DO 100 I=2,10
                  IF(.NOT.A(1,I).EQ.9999.0) GO TO 70
C                       IF(SUPRES(J,K,N).EQ.1) WRITE(6,1004) IV,A(1,I-1)
                        TABLE2=A(LI,I-1)+CP*(A(UI,I-1)-A(LI,I-1))
                        RETURN
C
```

```
   70          IF(IV-A(1,I)) 80,90,100
C              IV < A(1,I)
   80                CV = (IV-A(1,I-1))/(A(1,I)-A(1,I-1))
                     LFT= A(LI,I-1)+CP*(A(UI,I-1)-A(LI,I-1))
                     RGT= A(LI,I)+CP*(A(LI,I)-A(LI,I))
                     TABLE2= LFT+CV*(RGT-LFT)
                     RETURN
C              IV = A(1,I)
   90                TABLE2=A(LI,I)+CP*(A(UI,I)-A(LI,I))
                     RETURN
C              END IF
  100      CONTINUE
C     END IF
C
C
      WRITE(6,1005)
      RETURN
C
C     FORMAT STATEMENTS FOR SUBROUTINE TABLE2**************************
C
 1001 FORMAT('0','SUBROUTINE TABLE2: INPUT PARAMETER BELOW DATA.'
     *        /'          INPUT PARAMETER                = ',F10.2,
     *        /'          USED LOWEST PARAMETER          = ',F10.2)
 1002 FORMAT('0','SUBROUTINE TABLE2: INPUT PARAMETER ABOVE DATA.'
     *        /'          INPUT PARAMETER                = ',F10.2,
     *        /'          USED HIGHEST PARAMETER         = ',F10.2)
 1003 FORMAT('0','SUBROUTINE TABLE2: INPUT INDEP. VAR. BELOW DATA.'
     *        /'          INPUT INDEP. VAR.              = ',F10.2,
     *        /'          USED LOWEST INDEP. VAR.        = ',F10.2)
 1004 FORMAT('0','SUBROUTINE TABLE2: INPUT INDEP. VAR. ABOVE DATA.'
     *        /'          INPUT INDEP. VAR.              = ',F10.2,
     *        /'          USED HIGHEST INDEP. VAR.       = ',F10.2)
 1005 FORMAT('0','SUBROUTINE TABLE2: ERROR. SUBROUTINE DID NOT END')
      END
C
C                                                              9-05-84
C *********************************************************************
      FUNCTION SUPRES(J,K,N)
C *********************************************************************
C     DETERMINES TO SUPPRESS WARNINGS FROM TABLE1 OR TABLE2 AFTER THEY
C     HAVE BEEN CALLED ABOUT 20 TIMES IN A ROW. (ABOUT 5 DT INTERVALS.)
C     *****************************************************************
      INTEGER J,K,N,SUPRES
C
      IF (K-J.EQ.1)GO TO 10
            N  = 0
            K  = 0
            J  = 0
            GO TO 15
```

```
   10 CONTINUE
         N   = N+1
         J   = K
   15 CONTINUE
C
      IF(N.EQ.20)WRITE(6,105)
  105 FORMAT('0','TABLE LOOKUP WARNINGS SUPPRESSED')
C
      IF(N.GE.20) SUPRES = 1
      IF(N.LT.20) SUPRES = 0
      RETURN
      END
C
C                                                             9-07-84
C ************************************************************************
      SUBROUTINE TGTNAV
C ************************************************************************
C     RNG, SCVV
C   **********************************************************************
C     NAVIGATES THE TARGET AND COMPUTES RELATIVE RANGE, RANGE RATES.
C     INCORPORATES ECM AND GLINT MODELS TO GIVE APPARENT TARGET
C     POSITIONS. ALSO CALCULATES LINE OF SIGHT ANGLES AND RATES.
C     ********************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOLT,PCOUNT,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     ******************************************
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPDT  ,NOLT  ,NPTS ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI    ,PII   ,MASS ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ   ,IA    ,IB   ,IC    ,GS    ,
     *           ID     ,IE    ,IF    ,IG    ,IH    ,II   ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2 ,SPAN  ,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /D/: MISSILE FLIGHT DYNAMICS PARAMETERS
C     ******************************************
C
      COMMON /D/ ALFA          ,BETA          ,VT            ,HMDOT         ,
     *           U             ,V             ,W             ,
     *           PHI           ,GAMMA         ,THETA         ,SY            ,
     *           CD            ,CY            ,CL            ,CR            ,
     *           CM            ,CN            ,P             ,Q             ,
```

```
     *             R              ,ALFADT        ,BETADT        ,PDOT          ,
     *             QDOT           ,RDOT          ,NZ            ,ALTUDE        ,
     *             XM             ,YM            ,XMDCT         ,YMDCT
C
C
C      ***********************************************
C      COMMON BLOCK /E/: INITIAL CONDITIONS
C      ***********************************************
C
       COMMON /E/ XM1             ,YM1           ,ALTUD1        ,U1            ,
     *             PITCH1         ,BANK1         ,HEDNG1        ,SIDES1        ,
     *             AOA1           ,PTCHR1        ,ROLLR1        ,YAWR1         ,
     *             XT1            ,YT1           ,HT1           ,TSPEED
C
C
C      ***********************************************
C      COMMON BLOCK /F/: GUIDANCE PARAMETERS
C      ***********************************************
C
       COMMON /F/ PH1             ,PH2           ,PH3           ,PH4           ,
     *             OFFSET         ,ALTATT        ,SGDZPU        ,MISDST        ,
     *             LAMDAZ         ,LAMDEL        ,KNFAZ         ,KNFEL         ,
     *             NZC            , PHIC         ,GAMMAC        ,PCLIM         ,
     *             PC             ,QC            ,RC            ,RANGE         ,
     *             SIGAZ          ,SIGEL         ,SIGDAF        ,SIGDEF        ,
     *             SYT            ,THETAT        ,XT            ,YT            ,
     *             HT             ,NYC           ,POPRNG
C
C
C      ***********************************************
C      COMMON BLOCK /H/: ECM/GLINT PARAMETERS
C      ***********************************************
C
       COMMON /H/ FREQ            ,SHIFTY        ,SHIFTH        ,BRNTHR        ,
     *             XECM           ,YECM          ,HECM          ,XGLNT         ,
     *             YGLNT          ,HGLNT         ,XTECM         ,YTECM         ,
     *             HTECM
C
C EXECUTABLE STATEMENTS:
C*****************************************************************
C
C *** TARGET MOTION
C
       XT = XT1
       YT = YT1+TSPEED*TIME
       HT = HT1
C
C      RELATIVE RANGE TO TARGET
```

```
C
      XR  =  XT-XM
      YR  =  YT-YM
      HR  =  HT-ALTUDE
C
C *** RANGE & MISS DISTANCE
C
      RNGE    =  SQRT(XR**2+YR**2+HR**2)
      IF(RNGE.GT.2000.)GO TO 25
            IF(RNGE.LT.RANGE)GO TO 25
                  CPA   = 1
                  MISDST     = RANGE
   25 CONTINUE
      RANGE      = RNGE
C
C
C *** GLINT MODEL (SUPPRESSED AT A RANGE OF 100 FT)
C ************** TESTPATCH TO SUPPRESS GLINT (KTEST = 1)
      KTEST = 1
      IF(KTEST.EQ.1) GO TO 50
C ************** ENDTEST
      CALL RNG(RAND)
C
      IF(PCGLNT.NE.1) GO TO 100
            IF(RANGE.LT.100.0) GO TO 50
                  XGLNT =  20*RAND
                  YGLNT =  50*RAND
                  HGLNT =  20*RAND
                  GO TO 100
   50       CONTINUE
                  XGLNT =  0.0
                  YGLNT =  0.0
                  HGLNT =  0.0
  100 CONTINUE
C
C
C *** ECM BLINKING MODEL (SUPPRESSED AT BURN-THROUGH RANGE)
C
C   * SELECT THE PHASE APPLIED TO THE ECM SIGNAL
C
      GO TO(101,102,103,104),NFAZE
  101       PHASE = 0.0
            GO TO 105
  102       PHASE = PI*0.5
            GO TO 105
  103       PHASE = PI
            GO TO 105
  104       PHASE = PI*1.5
```

```
  105 CONTINUE
C
      POPRNG = PHASE*PII
      IF(RANGE.LT.BRNTHR) GO TO 350
            XECMW = SIN(2*PI*FREQ*TIME+PHASE)
            XECM  = SQWV(XECMW,SHIFTX)
C
            YECMW = SIN(2*PI*FREQ*TIME+PHASE)
            YECM  = SQWV(YECMW,SHIFTY)
C
            HECMW = SIN(2*PI*FREQ*TIME+PHASE)
            HECM  = SQWV(HECMW,SHIFTH)
            GO TO 400
  350 CONTINUE
            XECM  = 0.0
            YECM  = 0.0
            HECM  = 0.0
  400 CONTINUE
C
      XTECM  = XT+XECM+XGLNT
      YTECM  = YT+YECM+YGLNT
      HTECM  = HT+HECM+HGLNT
C
C
C *** RELATIVE RANGE AND RANGE RATE TO RADAR TARGET
C
      XRECM  = XTECM-XM
      YRECM  = YTECM-YM
      HRECM  = HTECM-ALTUDE
      RNGECM = SQRT(XRECM**2+YRECM**2)
      RGECMT = SQRT(XRECM**2+YRECM**2+HRECM**2)
C
      XDCTR  = -XMDOT
      YDCTR  = TSPEED-YMDOT
      HDCTR  = -HMDOT
      HORDTR = SQRT(XDOTR**2+YDOTR**2)
C
C
C *** SEEKER LOS AND LOS RATE CALCULATIONS
C
      SYT    = ATAN2(YRECM,XRECM)
      TRAKAZ = ATAN2(YMDOT,XMDOT)
      SIGAZ  = SYT-TRAKAZ
C
      VTANAZ = -XDOTR*SIN(SYT)+YDOTR*COS(SYT)
      SIGDAZ = VTANAZ/RNGECM
      SIGDAF = REALPL(SIGDAF,KNFAZ,SIGDAZ,DT)
C
```

```
      THETAT  = ATAN2(HRECM,RNGECM)
      HORDOT  = SQRT(XMDOT**2+YMDOT**2)
      TRAKEL  = ATAN2(HMDOT,HORDOT)
      SIGEL   = THETAT-TRAKEL
C
      VTANEL  = HDOTR*COS(THETAT)+HORDTR*SIN(THETAT)
      SIGDEL  = VTANEL/RGECMT
      SIGDEF  = REALPL(SIGDEF,KNFEL,SIGDEL,DT)
C
C
      RETURN
      END
C
C                                                                  8-01-84
C **********************************************************************
      SUBROUTINE RNG(RAND)
C **********************************************************************
C     GENERATES A RANDOM NUMBER BETWEEN -1.0 & 1.0
C     ******************************************************************
C
      IMPLICIT REAL(A-Z)
      DATA SEED/4.0/
C
      RAND = (SEED + 3.14159)**5.04
      RAND = (RAND - IFIX(RAND)-0.5)*2.0
      SEED = RAND
C
      RETURN
      END
C
C                                                                  9-07-84
C **********************************************************************
      FUNCTION SQWV(WAVE,AMPL)
C **********************************************************************
C     GENERATES A SQUAREWAVE FROM A SINWAVE
C     ******************************************************************
C
      IMPLICIT REAL(A-Z)
C
      IF (WAVE.GT.0.0) SQWV=AMPL
      IF (WAVE.LT.0.0) SQWV=-AMPL
      IF (WAVE.EQ.0.0) SQWV= 0.0
C
      RETURN
      END
C
C                                                                  9-07-84
C **********************************************************************
```

```
      SUBROUTINE PREPAR
C     ****************************************************************
C     DESIGNATES UP TO 20 PARAMETERS TO BE STORED FOR OUTPUT AND
C     ASSIGNS THEM TO THE NEXT ROW OF THE PTS ARRAY. EACH PARAM. IS
C     STORED IN A COLUMN OF THE ARRAY FROM ROW 3 UNTIL THE END OF THE
C     ARRAY IS REACHED OR CPA OCCURS. ROWS 1 & 2 CONTAIN MINIMUM AND
C     MAXIMUM VALUES OF EACH VARIABLE RESPECTIVELY. COL. 1 CONTAINS
C     THE VALUES OF TIME FOR EACH POINT. NPTS IS COMMON TO BLOCK A
C     AND INDICATES THE NUMBER OF POINTS STORED.
C     ****************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOUT,PCOUNT,NFAZE
      DIMENSION KEEP(20)
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPDT  ,NOUT  ,NPTS  ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI    ,PII   ,MASS  ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ   ,IA    ,IB    ,IC    ,QS    ,
     *           ID     ,IE    ,IF    ,IG    ,IH    ,II    ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2,SPAN  ,NFAZE
C
C
      COMMON /C/ KPTCHR        ,KROLLR        ,KYAWRT        ,KBANK        ,
     *           KGAMMA        ,KALT          ,CGARML        ,CGARMN       ,
     *           RRTLIM        ,PLIM          ,KNY           ,KNZ          ,
     *           AILRON        ,STBLTR        ,RUDDER        ,
     *           BSERO         ,NZSERO        ,NYSERO
C
C
      COMMON /D/ ALFA          ,BETA          ,VT            ,HMDOT        ,
     *           U             ,V             ,W             ,
     *           PHI           ,GAMMA         ,THETA         ,SY           ,
     *           CD            ,CY            ,CL            ,CR           ,
     *           CM            ,CN            ,P             ,Q            ,
     *           R             ,ALFADT        ,BETADT        ,PDOT         ,
     *           QDOT          ,RDOT          ,NZ            ,ALTUDE       ,
     *           XM            ,YM            ,XMDOT         ,YMDOT
C
      COMMON /E/ XM1           ,YM1           ,ALTUD1        ,U1           ,
     *           PITCH1        ,BANK1         ,HEDNG1        ,SIDES1       ,
     *           AOA1          ,PTCHR1        ,ROLLR1        ,YAWR1        ,
     *           XT1           ,YT1           ,HT1           ,TSPEED
C
      COMMON /F/ PH1           ,PH2           ,PH3           ,PH4          ,
     *           OFFSET        ,ALTATT        ,SGDZPU        ,MISDST       ,
     *           LAMDAZ        ,LAMDEL        ,KNFAZ         ,KNFEL        ,
     *           NZC           , PHIC         ,GAMMAC        ,PCLIM        ,
     *           PC            ,QC            ,RC            ,RANGE        ,
```

```
     *              SIGAZ          ,SIGEL         ,SIGDAF        ,SIGDEF       ,
     *              SYT            ,THETAT        ,XT            ,YT           ,
     *              HT             ,NYC           ,POPRNG
C
      COMMON /G/    ADA            ,SIDESL        ,BANK          ,FLTPHC       ,
     *              BANKC          ,PITCH         ,ROLLRT        ,ROLRTC       ,
     *              PTCHRT         ,YAWRT         ,HEADNG        ,FLTPTH       ,
     *              HEADT          ,ELEVT         ,DSIGAZ        ,DSIGEL       ,
     *              DSGDAZ         ,DSGDEL        ,ERFEK         ,ERFRR        ,
     *              ERFAZ          ,ERFEL
C
      COMMON /H/    FREQ           ,SHIFTY        ,SHIFTH        ,BRNTHR       ,
     *              XECM           ,YECM          ,HECM          ,XGLNT        ,
     *              YGLNT          ,HGLNT         ,XTECM         ,YTECM        ,
     *              HTECM
C
C
      COMMON /I/    PTS(300,20),PLTN(6,7),XN(6,7),YN(6,7),TITLE(6),
     *              LEG(4,20)
C
C  EXECUTABLE  STMTS  ****************************************************
C
C       RADIAN TO DEGREE CONVERSIONS FOR OUTPUT
C
      ADA    = ALFA*PII
      SIDESL = BETA*PII
      BANK   = PHI*PII
      FLTPHC = GAMMAC*PII
      BANKC  = PHIC*PII
      PITCH  = THETA*PII
      ROLLRT = P*PII
      ROLRTC = PCLIM*PII
      PTCHRT = Q*PII
      YAWRT  = R*PII
      HEADNG = SY*PII
      FLTPTH = GAMMA*PII
      HEADT  = SYT*PII
      ELEVT  = THETAT*PII
      DSIGAZ = SIGAZ*PII
      DSIGEL = SIGEL*PII
      DSGDAZ = SIGDAF*PII
      DSGDEL = SIGDEF*PII
C
C  *** CREATE THE MISSION PHASE MARKER (MARK)
C
      MARK = 0.0
      IF(PH1.EQ.1)MARK = 1.0
      IF(PH2.EQ.1)MARK = 2.0
```

```
      IF(PH3.EQ.1)MARK =  3.0
      IF(PH4.EQ.1)MARK =  4.0
C
      NPTS = NPTS+1
      K    = NPTS+2
C
C *** COMPUTE THE ERROR FUNCTIONS
C
      IF(MARK.EQ.4) GO TO 50
            ERRBK     = 0.0
            ERRRR     = 0.0
            ERRAZ     = 0.0
            ERREL     = 0.0
            GO TO 100
   50 CONTINUE
            ERRBK     = ERRBK + ABS(BANKC-BANK)    *DT
            ERRRR     = ERRRR + ABS(RCLRTC-ROLLRT)*DT
            ERRAZ     = ERRAZ + ABS(DSGDAZ)        *DT
            ERREL     = ERREL + ABS(DSGDEL)        *DT
  100 CONTINUE
C
      ERFBK      = ERRBK/TIME
      ERFRR      = ERRRR/TIME
      ERFAZ      = ERRAZ/TIME
      ERFEL      = ERREL/TIME
C
C *** SELECT THE VARIABLES TO BE STORED
C
      KEEP(1)  = TIME
C
C     GRAPH 1
      KEEP(2)  = NZC
      KEEP(3)  = NZ
C
C     GRAPH 2
      KEEP(4)  = BANKC
      KEEP(5)  = BANK
C
C     GRAPH 3
      KEEP(6)  = ROLRTC
      KEEP(7)  = ROLLRT
C
C     GRAPH 4
      KEEP(8)  = YECM
      KEEP(9)  = YCLNT
C
C     GRAPH 5
      KEEP(10) = AILRCN
```

```
      KEEP(11) = STBLTR
      KEEP(12) = RUDDER
C
C     GRAPH 6
      KEEP(13) = ALTUDE
C
C     GRAPH 7
      KEEP(14) = XM
      KEEP(15) = YM
      KEEP(16) = XT
      KEEP(17) = YT
C
C     SPARES
      KEEP(18) = RANGE
      KEEP(19) = MARK
      KEEP(20) = NYC
C
C *** STORE MINIMUM AND MAXIMUM VALUES OF EACH VARIABLE
C
      DO 20 I=1,20
          IF (NPTS.GT.1) GO TO 10
              PTS(1,I)  = KEEP(I)
              PTS(2,I)  = KEEP(I)
   10     CONTINUE
          PTS(1,I)  = AMIN1(PTS(1,I),KEEP(I))
          PTS(2,I)  = AMAX1(PTS(2,I),KEEP(I))
   20 CONTINUE
C
C *** STORE VALUES OF EACH VARIABLE WITH THE TIME
C
      DO 30 I=1 ,20
          PTS(K,I)  = KEEP(I)
   30 CONTINUE
C
C *** CHECK FOR ARRAY OVERFLOW AND TERMINATE AT 295 POINTS
C
      IF(NPTS.GE.295) CPA = 3.0
C
C
      RETURN
      END
C
C                                                                  9-07-84
C ***********************************************************************
      SUBROUTINE OUTPUT(NPTS,CPA)
C ***********************************************************************
C     HEADER, PLOT1, PLOT2, PLOT21, PLOT3
C    *********************************************************************
```

```
C       RECEIVES THE DATA FROM THE PROGRAM IN THE PTS ARRAY. PRINTS THE
C       PRIME OUTPUT AND DATA RANGES TO THE SCREEN AND TABULAR OUTPUT TO
C       FILE 9 (TCMC DATA). THEN IT PROCESSES THE DATA FOR PLOTTING USING
C       DISSPLA. THE BASIC FORMAT OF THE SEVEN GRAPHS CALLED IN THIS S/R
C       IS FIXED BUT THE TITLES AND THE VARIABLES CAN BE EASILY CHANGED BY
C       CHANGING THE PARAMETERS IN THE DATA STATEMENTS IN THIS S/R. THE
C       TITLE GIVEN HERE WILL BE PRINTED FOR ALL PLOTS. @
C       ****************************************************************
        IMPLICIT REAL(A-Z)
        INTEGER  KFILE,NPTS,I,J,K,L,M,N,NN,CPA,
     *           NDV(5,7),DV(4),LEG(4,20),PLTN(6,7),
     *           XN(6,7),YN(6,7),TITLE(8,4),MESS1(2),MESS2(2)
C
        DIMENSION YP(10),XP(10)
C
C       ****************************************************************
C
        COMMON /F/ PH1          ,PH2          ,PH3          ,PH4          ,
     *             OFFSET       ,ALTATT       ,SGDZPU       ,MISDST       ,
     *             LAMDAZ       ,LAMDEL       ,KNFAZ        ,KNFEL        ,
     *             NZC          , PHIC        ,GAMMAC       ,PCLIM        ,
     *             PC           ,QC           ,RC           ,RANGE        ,
     *             SIGAZ        ,SIGEL        ,SIGDAF       ,SIGDEF       ,
     *             SYT          ,THETAT       ,XT           ,YT           ,
     *             HT           ,NYC          ,POPRNG
C
        COMMON /G/ AOA          ,SIDESL       ,BANK         ,FLTPHC       ,
     *             BANKC        ,PITCH        ,ROLLRT       ,ROLRTC       ,
     *             PTCHRT       ,YAWRT        ,HEADNG       ,FLTPTH       ,
     *             HEADT        ,ELEVT        ,DSIGAZ       ,DSIGEL       ,
     *             DSGDAZ       ,DSGDEL       ,ERFEK        ,ERFRR        ,
     *             ERFAZ        ,ERFEL
C
        COMMON /H/ FREQ         ,SHIFTY       ,SHIFTH       ,BRNTHR       ,
     *             XECM         ,YECM         ,HECM         ,XGLNT        ,
     *             YGLNT        ,HGLNT        ,XTECM        ,YTECM        ,
     *             HTECM
C
        COMMON /I/ PTS(300,20)
C
C
C   ********************************************************************
C   ************ INITIALIZE THE GRAPHICS CONTROL ARRAY **************
C
C           EACH ROW IN THE DATA STATEMENT (COLUMN OF THE ARRAY)
C           CORRESPONDS TO A SINGLE GRAPH. THE FIRST ELEMENT IS
C           A FLAG (1 TO PLOT, 0 TO PASS OVER). THE NEXT 4 ELEMENTS
```

```
C         ARE THE NUMBERS (BETWEEN 1 AND 20) OF THE DEPENDENT
C         VARIABLES TO BE PLOTTED AGAINST TIME.
C
C
      DATA    NOV      /  0   ,2   ,3   ,20  ,0   ,
     *                    0   ,4   ,5   ,0   ,0   ,
     *                    0   ,6   ,7   ,0   ,0   ,
     *                    0   ,8   ,9   ,0   ,0   ,
     *                    0   ,10  ,11  ,12  ,0   ,
     *                    0   ,13  ,0   ,0   ,0   ,
     *                    0   ,0   ,0   ,0   ,0   /
C
C *** LOAD THE MESSAGES AND VARIABLES TO APPEAR IN THE UPPER
C     LEFT HAND CORNER OF EACH GRAPH.
C
      DATA    MESS1   /'FREQ  = '/
      DATA    MESS2   /'PHASE = '/
C
C EXECUTABLE STMTS:
C
      VAR1  = FREQ
      VAR2  = PUPRNG
C
C ***********************************************************************
C *******  READ TITLE CAPTIONS FOR GRAPHS AND PRINTOUT *************
C                 FROM FILE NO.2: LABELS DATA
C
      IF(FLAG.EQ.1.0) GO TO 18
C
C *** READ IN OVERALL TITLE LINES (4 LINES OF 32 CHARACTERS)
C
      READ(2,5)((TITLE(J,I),J=1,8),I=1,4)
    5 FORMAT(20X,8A4)
C
C *** READ IN EACH GRAPH TITLE & ITS AXIS LABELS (24 CHARACTERS EACH)
C
      DO 10 J=1,7
            READ(2,12)(PLTN(I,J),I=1,6)
            READ(2,12)(XN(I,J),I=1,6)
            READ(2,12)(YN(I,J),I=1,6)
   10 CONTINUE
   12 FORMAT(20X,6A4)
C
C *** READ IN THE LEGEND LABEL FOR EACH OF THE 20 STORED VARIABLES
C     (16 CHARACTERS EACH)
C
      READ(2,15)((LEG(I,J),I=1,4),J=1,20)
   15 FORMAT(20X,4A4)
```

```
C
      FLAG = 1.0
   18 CONTINUE
C
C
C **************************************************************************
C ***************** WRITE PRIMARY DATA OUTPUT ***********************
C          TO FILE 6 (TERMINAL) AND THEN FILE 9 (TCMC DATA)
C
      DO 100 K=1,2
         IF(K.EQ.1)KFILE=6
         IF(K.EQ.2)KFILE=9
C
C ***    TITLES
C
         WRITE(KFILE,20)((TITLE(I,J),I=1,8),J=1,4)
   20    FORMAT('1',4(20X,8A4/)//)
C
C *** INDICATE HOW THE SIMULATION TERMINATED
C
         GO TO(30,40,50),CPA
C
   30       WRITE(KFILE,35)
   35       FORMAT(1X, 'SIMULATION TERMINATED DUE TO CPA')
            GO TO 60
C
   40       WRITE(KFILE, 45)
   45       FORMAT(1X, 'SIMULATION TERMINATED DUE TO FINTIME')
            GO TO 60
C
   50       WRITE(KFILE, 55)
   55       FORMAT(1X, 'SIMULATION TERMINATED DUE TO FULL ARRAY')
C
   60    CONTINUE
C
C ***    VALUE OF THE ITERATED PARAMETER
C
         WRITE(KFILE,65) FREQ
   65    FORMAT(1X,'*** BLINKER FREQUENCY=  ',F8.2)
C
         WRITE(KFILE,67) PCPRNG
   67    FORMAT(1X,'*** BLINKER PHASE     =  ',F6.0//)
C
C ***    LIST THE PRIMARY DATA OUTPUTS
C
         WRITE(KFILE,75) MISDST,ERFBK,ERFRR,ERFAZ,ERFEL
   75    FORMAT(1X,'      MISS     * ==================== ERROR FUNC
     *TIONS =====================*'/'     DISTANCE   *     BANK       *
```

```
     *   ROLLRATE   *     AZIMUTH     *    ELEVATION    *'//
     *   5(3X,F10.5,2X,'*')//////)
C
C ***       LIST THE VARIABLE RANGES
C
          WRITE(KFILE,80)
   80     FORMAT(18X,'******* RANGES FOR ALL SAVED VARIABLES *******'
     *        ///30X, '      MINIMUM             MAXIMUM   '//)
          WRITE(KFILE,90)(((LEG(I,J),I=1,4), PTS(1,J),PTS(2,J)),J=1,20)
   90     FORMAT(11X, 4A4, 4X, F12.6, 3X, F12.6)
C
  100 CONTINUE
C
C
C *****************************************************************
C ****************** TABULAR DATA OUTPUT **************************
C
      DO 200 I=1,4
          J=(I-1)*4+2
          K=J+3
          WRITE(9,20)((TITLE(L,M),L=1,8),M=1,4)
          WRITE(9,65) FREQ
          WRITE(9,125) I
  125     FORMAT(50X,'DATA SET NUMBER  ',I1,'  OF 4'//)
          WRITE(9,135)(LEG(L,1),L=1,4),((LEG(L,M),L=1,4),M=J,K)
  135     FORMAT(2X, 20A4///)
C
          NN=NPTS+2
          DO 150 N=3,NN
C               START NEW PAGE EVERY 65 LINES
              L = MOD(N,65)
              IF(L.NE.0) GO TO 140
                  WRITE(9,20)((TITLE(L,M),L=1,8),M=1,4)
                  WRITE(9,65) FREQ
                  WRITE(9,125) I
                  WRITE(9,135)(LEG(L,1),L=1,4),
     *                       ((LEG(L,M),L=1,4),M=J,K)
  140         CONTINUE
              WRITE(9,145) PTS(N,1),(PTS(N,L),L=J,K)
  145         FORMAT(1X,5('*',F12.6,3X))
  150     CONTINUE
  200 CONTINUE
C
C
C *****************************************************************
C ******* PRODUCE GRAPHIC OUTPUT USING DISSPLA SUBROUTINES *********
C
C
```

```
C
C     *************
C     GRAPH NC.S 1-6
C     *************
C
      DO 300 J = 1,6
C           IF((J.EQ.4).AND.(FREQ.EQ.0.0))GO TC 28C
            IF(NDV(1,J).EC.0)GO TC 280
                  DC 250 I = 2,5
                        DV(I-1)    = NDV(I,J)
  25C             CONTINUE
C
                  CALL  PLOT1(MESS1,MESS2,VAR1,VAR2)
                  CALL  HEADER(J,TITLE,PLTN,XN,YN)
                  CALL  PLOT2(PTS,LEG,DV,1,NPTS)
C
  28C       CCNTINUE
  300 CONTINUE
C
C
C     ********
C     GRAPH NC.7 *** 3-DIMENSIONAL GEO PLCT
C     ********
C
      IF(NDV(1,7).EQ.0) GC TC 900
C
      CALL PLCT1(MESS1,MESS2,VAR1,VAR2)
      CALL PLCT3(NPTS,PTS,TITLE)
C
C ************
C
 900  CONTINUE
C
      RETURN
      END
C
C                                                                  9-07-84
C *********************************************************************
      SUBROUTINE HEADER(KP,TITLE,PLTN,XN,YN)
C *********************************************************************
C     LOADS THE 1D ARRAYS USED TO PASS LABELS TO DISSPLA FROM THE
C     2D ARRAYS PACKED WITH THE TITLE OF EACH OF THE SEVEN GRAPHS AND
C     THE AXIS LABELS. CALLS THE DISSPLA HEADING AND AXIS LABEL
C     *****************************************************************
C
      INTEGER KP,PLTN(6,7),XN(6,7),YN(6,7),XNM(6),YNM(6),HD(6),
     *        1, TITLE(8,4)
C
```

```
      DO 50 J=1,3
         DO 30 I=1,8
            HD(I)       = TITLE(I,J)
30       CONTINUE
         CALL HEADIN(HD   ,32    ,   1.1   ,4)
50    CONTINUE
      DO 60 I=1,6
         HD(I)    = PLTN(I,KP)
         XNM(I)    = XN(I,KP)
         YNM(I)    = YN(I,KP)
60    CONTINUE
      CALL HEADIN(HD    ,24    ,   1.0    ,4)
C
      CALL XNAME( XNM   ,24)
      CALL YNAME( YNM   ,24)
C
      RETURN
      END
C ********************************************************************
C     FILE: PLOT1
C ********************************************************************
C     CONTAINS ALL S/R'S THAT INTERFACE WITH DISSPLA EXCEPT HEADER
C     ****************************************************************
C
C                                                             9-10-84
C ********************************************************************
      SUBROUTINE PLOT1(MESS1,MESS2,VAR1,VAR2)
C ********************************************************************
C     INITIALIZES DISSPLA FOR A NEW PAGE AND GRAPH
C     ****************************************************************
C
      IMPLICIT REAL(A-H,O-Z),INTEGER(I-N)
      DIMENSION MESS1(2),MESS2(2)
C
      CALL NOCHEK
      CALL GRACE(0.)
      CALL BLOWUP(0.647)
      CALL PAGE(11.,8.5)
      CALL HWROT('AUTO')
      CALL HWSCAL('SCREEN')
C     CALL NOEROR
      CALL PHYSOR(1.,.75)
      CALL AREA2D(9.,6.5)
C     CALL SWISSM
C
C *** PUT THE MESSAGES INTO THE GRAPHS
C
      CALL MESSAG(MESS1,8,0.2,6.0)
```

```
      CALL REALNC(VAR1 ,2,'ABUT','ABUT')
      CALL MESSAG(MESS2,8,0.2,5.6)
      CALL REALNC(VAR2 ,0,'ABUT','ABUT')
      CALL BLREC(0.1,5.5,1.8,0.8,0.01)
C
      RETURN
      END
C
C                                                                 8-26-84
C *********************************************************************
      SUBROUTINE PLOT2(PTS,LEG,DV,IV,NPTS)
C *********************************************************************
C     LOADS THE LEGEND ARRAY WITH CURVE LABELS; GETS THE POINTS FOR
C     EACH CURVE FROM THE PTS ARRAY; PLOTS EACH CURVE. IV,DV ARE IN-
C     DEPENDENT, DEPENDENT VARIABLE INDEXES. THEY GIVE THE COLUMN IN
C     THE PTS ARRAY WHERE THEIR VALUES ARE FOUND. LEG IS THE PACKED
C     ARRAY CONTAINING LEGEND LABELS FOR EACH VARIABLE. DV IS A 4
C     ELEMENT ARRAY CONTAINING THE INDEXES FOR UP TO 4 DEP VAR'S.
C     *****************************************************************
C
      IMPLICIT  REAL(A-H,O-Z),INTEGER(I-N)
      INTEGER   DV(4),IPACK(100),LEG(4,20),LBL(5)
      DIMENSION PTS(300,20), XP(300), YP(300)
C
      DATA MONEY /'$    '/
C
C
C *** FIRST FIND OUT HOW MANY CURVES THERE ARE
C
      I=1
    5 IF (DV(I).EQ.0)GO TO 10
            MCRV = I
            IF (I.EQ.4)GO TO 10
            I    = I+1
            GO TO 5
   10 CONTINUE
C
C *** NEXT PACK THE LEGEND ARRAY AND RANGE THE Y-AXIS
C
      DO 30 I=1,MCRV
            K    = DV(I)
            IF(I.GT.1) GO TO 15
                  YMIN = PTS(1,K)
                  YMAX = PTS(2,K)
                  GO TO 20
   15       CONTINUE
                  YMIN = AMIN1(YMIN,PTS(1,K))
                  YMAX = AMAX1(YMAX,PTS(2,K))
```

```
   20       CONTINUE
C
            DO 25 J=1,4
                  LBL(J)      = LEG(J,K)
   25       CONTINUE
            LBL(5) = MONEY
            CALL LINES(LBL,IPACK,I)
   30 CONTINUE
C
C *** RANGE THE INDEPENDENT VARIABLE(S)
C
      XMIN = PTS(1,IV)
      XMAX = PTS(2,IV)
C
C *** NOW PLOT THE CURVES
C
      CALL GRAF(XMIN,'SCALE',XMAX,YMIN,'SCALE',YMAX)
      DO 90 I=1,MCRV
            DO 80 J=1,NPTS
                  XP(J)       = PTS(J+2,IV)
                  YP(J)       = PTS(J+2,DV(I))
   80       CONTINUE
            CALL CURVE(XP,YP,NPTS,1)
   90 CONTINUE
C
C PLOT THE GRID & LEGEND AND FINISH THE GRAPH
C
      CALL PLOT21(IPACK,MCRV)
C
      RETURN
      END
C
C                                                                   7-25-84
C ******************************************************************************
      SUBROUTINE PLOT21(IPACK,MCRV)
C ******************************************************************************
C     PLOTS THE LEGEND & THE GRID AND ENDS THIS GRAPH
C     *************************************************************************
C
      IMPLICIT INTEGER(I-N)
      DIMENSION IPACK(100)
C
      XW=XLEGND(IPACK,MCRV)
      YW=YLEGND(IPACK,MCRV)
      CALL LEGEND(IPACK,MCRV,0.5,0.5)
      CALL BLREC(0.4,0.4,XW+0.2,YW+0.2,.01)
      CALL DOT
```

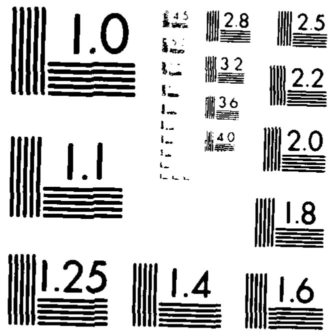
AD-A152 193 AN INVESTIGATION INTO THE CONTROL LIMITATIONS OF A BANK TO TURN MISSILE IN THE TERMINAL HOMING PHASE(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA B P ANDERSON SEP 84 3/3

UNCLASSIFIED F/G 17/7 NL

END

FILMED

DTIC


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

```
      CALL GRID(1,1)
      CALL RESET('DOT')
      CALL ENDPL(0)
      RETURN
      END
C
C                                                                09-05-84
C ************************************************************************
      SUBROUTINE PLOT3(NPTS,PTS,TITLE)
C     ********************************************************************
C     CREATES A 3-D GEOGRAPHICAL PLOT OF MISSILE AND TARGET
C     TRACKS. BY CHANGING THE VIEWPOINT YOU CAN OBSERVE DIFFERENT
C     PORTIONS OF THE MISSION FROM VARIOUS PERSPECTIVES.
C     ********************************************************************
      IMPLICIT REAL(A-H,O-Z), INTEGER(I-N)
      INTEGER TITLE(8,4),HD(8)
      DIMENSION PTS(300,20),XP(300),YP(300),ZP(300),IPACK(100)
C
C     CALL BLOWUP(1.0)
C
C *** CREATE THE TITLES
C
      DO 40 J=1,3
          DO 30 I=1,8
             HD(I)      = TITLE(I,J)
   30     CONTINUE
          CALL HEADIN(HD   ,32   ,  1.1   ,4)
   40 CONTINUE
      CALL HEADIN('GEOGRAPHICAL TRACKS',19,1.0,4)
C
C *** DEFINE THE WORKBOX
C
      CALL VOLM3D(10,20,7.5)
C
C *** LABEL THE AXES
C
      CALL X3NAME('FEET EAST',9)
      CALL Y3NAME('FEET NORTH',10)
      CALL Z3NAME('ALTITUDE (FT)',13)
C
C *** DEFINE THE VIEWPOINT
C
      CALL VUANGL(130,40,26)
C
C *** SET UP THE AXIS SYSTEM
C
      CALL CROSS
      CALL GRAF3D(-1000.,1000.,4000.,   0.0,5000.,25000.,
```

```
     *              0.0,100.0,300.0)
C
C *** DRAW THE MISSILE'S TRACK
C
      IX    = 15
      IY    = 14
      IZ    = 13
C
      DO 50 J=1,NPTS
              XP(J)  = PTS(J+2,IX)
              YP(J)  = PTS(J+2,IY)
              ZP(J)  = PTS(J+2,IZ)
   50 CONTINUE
      CALL CURV3D(XP,YP,ZP,NPTS,2)
C
C *** NOW PLOT THE TARGET'S TRACK
C
      IX    = 17
      IY    = 16
      IZ    =  0
C
      DO 60 J=1,NPTS
              XP(J)  = PTS(J+2,IX)
              YP(J)  = PTS(J+2,IY)
              ZP(J)  = 10.0
   60 CONTINUE
      CALL CURV3D(XP,YP,ZP,NPTS,5)
C
      CALL ENDPL(0)
      RETURN
      END
*************************************************************************
```

```
                         FILE #2 CONTENTS
              CONTAINS ALL TITLES AND LABELS USED

GLOBAL TITLE LINES:*      CRUISE MISSILE TESTS       *
                   *  SEA SKIMMER (BTT) - NO POPUP   *
                   *    HI-FREQ SCANS - 0.0-30.0     *
                   *            9-17-84              *
PLOT & AXIS LABELS:*   NORMAL LOAD FACTOR     *    *************
            X..... *          TIME            *        GRAPH 1
            Y..... *  CMD & ACTUAL NZC, NZY   *    *************
                   *    BANK ANGLE CONTROL    *    *************
            X..... *          TIME            *        GRAPH 2
            Y..... *  COMMANDED & ACTUAL PHI  *    *************
                   *     ROLL RATE CONTROL    *    *************
```

```
                X.....  :         TIME              :     GRAPH 3
                Y.....  :  COMD & ACTUAL ROLL RATE  :  *************
                        :   ECM & GLINT SIGNALS     :  *************
                X.....  :         TIME              :     GRAPH 4
                Y.....  :   ECM & GLINT Y-SHIFT     :  *************
                        :    FLIGHT CONTROLS        :  *************
                X.....  :         TIME              :     GRAPH 5
                Y.....  : 3 PRIMARY FLT. CONTROLS   :  *************
                        :   ALTITUDE CONTROL        :  *************
                X.....  :         TIME              :     GRAPH 6
                Y.....  :    ACTUAL ALTITUDE        :  *************
                        :                           :  *************
                X.....  :                           :     BLANK
                Y.....  :                           :  *************
LEGENDS:        1.      :TIME.......(SEC)  :   *********************
                2.  G1. :NZC..........(G)  :   PARAMETERS STORED BY
                3.      :NZ...........(G)  :          PREPARE
                4.  G2. :BANKC......(DEG)  :   *********************
                5.      :BANK.......(DEG)  :
                6.  G3. :R/RT CMD...(DPS)  :
                7.      :ROLL RATE..(DPS)  :
                8.  G4. :ECM SHIFT...(FT)  :
                9.      :GLINT SHIFT(FT)   :
                10. G5. :STBLTR.....(DEG)  :
                11.     :AILRON.....(DEG)  :
                12.     :RUDDER.....(DEG)  :
                13. G6. :ALTITUDE....(FT)  :
                14. XX. :XM....(FT NORTH)  :
                15.     :YM.....(FT EAST)  :
                16.     :XT....(FT NORTH)  :
                17.     :XM.....(FT EAST)  :
                18.     :RANGE.......(FT)  :
                19.     : PHASE MARKER     :
                20.     :NYC..........(G)  :
```

APPENDIX E

LISTING FOR SUBROUTINE MISSN1

```
C                                                                9-07-84
C *************************************************************************
      SUBROUTINE MISSN1
C *************************************************************************
C     SEA SKIMMER GUIDANCE SCHEME. ALLOWS VARIABLE POPUP IN CLOSE
C     WITH NO OFFSET TURN. AFTER POPUP, GUIDANCE IS STANDARD PRO-
C     PORTIONAL. POPUP USES ONLY ALTITUDE COMMAND VICE GAMMAC.
C
C     MAKES MISSION PHASE DECISIONS AND INVOKES THE DIFFERENT MODES
C     OF GUIDANCE AS REQUIRED. DELIVERS NZC AND PHIC TO THE AUTOPILOT
C     CONTROL LOOPS. NYC IS ASSUMED TO ALWAYS BE ZERO. NZC IS LIMITED
C     TO +4.0 AND -2.0 G'S.
C     *********************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NCUT,PCOUNT,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /A/: MISCELLANEOUS CONSTANTS
C     ******************************************
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPDT  ,NCUT  ,NPTS ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI    ,PII   ,MASS ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ   ,IA    ,IB   ,IC    ,GS    ,
     *           ID     ,IE    ,IF    ,IG    ,IH    ,II   ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2 ,SPAN  ,NFAZE
C
C
C     ******************************************
C     COMMON BLOCK /C/: CONTROL SYSTEM PARAMETERS
C     ******************************************
C
      COMMON /C/ KPTCHR        ,KROLLR        ,KYAWRT        ,KBANK        ,
     *           KGAMMA        ,KALT          ,CGARML        ,CGARMN       ,
     *           RRTLIM        ,PLIM          ,KNY           ,KNZ          ,
     *           AILRON        ,STBLTR        ,RUDDER        ,
     *           BSERO         ,NZSERO        ,NYSERO
C
C
C     ******************************************
C     COMMON BLOCK /D/: MISSILE FLIGHT DYNAMICS PARAMETERS
```

```
C      *******************************************
C
      COMMON /D/ ALFA          ,BETA         ,VT          ,HMDOT        ,
     *           U             ,V            ,W           ,
     *           PHI           ,GAMMA        ,THETA       ,SY           ,
     *           CD            ,CY           ,CL          ,CR           ,
     *           CM            ,CN           ,P           ,Q            ,
     *           R             ,ALFADT       ,BETADT      ,PDOT         ,
     *           QDOT          ,RDOT         ,NZ          ,ALTUDE       ,
     *           XM            ,YM           ,XMDOT       ,YMDOT
C
C      *******************************************
C      COMMON BLOCK /F/: GUIDANCE PARAMETERS
C      *******************************************
C
      COMMON /F/ PH1           ,PH2          ,PH3         ,PH4          ,
     *           OFFSET        ,ALTATT       ,SGDZPU      ,MISDST       ,
     *           LAMDAZ        ,LAMDEL       ,KNFAZ       ,KNFEL        ,
     *           NZC           , PHIC        ,GAMMAC      ,PCLIM        ,
     *           PC            ,QC           ,RC          ,RANGE        ,
     *           SIGAZ         ,SIGEL        ,SIGDAF      ,SIGDEF       ,
     *           SYT           ,THETAT       ,XT          ,YT           ,
     *           HT            ,NYC          ,POPRNG
C
C      *******************************************
C      COMMON BLOCK /G/: OUTPUT PARAMETERS
C      *******************************************
C
      COMMON /G/ ADA           ,SIDESL       ,BANK        ,FLTPHC       ,
     *           BANKC         ,PITCH        ,ROLLRT      ,ROLRTC       ,
     *           PTCHRT        ,YAWRT        ,HEADNG      ,FLTPTH       ,
     *           HEADT         ,ELEVT        ,DSIGAZ      ,DSIGEL       ,
     *           DSGDAZ        ,DSGDEL       ,ERFBK       ,ERFRR        ,
     *           ERFAZ         ,ERFEL
C
C EXECUTABLE STMTS:
C*************************************************************************
C
C      SET POPUP RANGE AND ALTITUDE
C
       POPRNG  =  15000.
C
C      MISSION PHASE LOGIC AND GUIDANCE COMMANDS
C
    1  IF(PH4.EQ.1) GO TO 40
       IF(PH1.EQ.1) GO TO 30
C
C ***  INGRESS FROM INITIAL CONDITION TO POPUP MANEUVER
```

```
C
      IF(RANGE.LT.POPRNG) GO TO 9
C
C        ALTITUDE HOLD
C
         ALTC   = 50.0
         ALTUDF = ALTUDE
         GAMMAC = KALT*(ALTC-ALTUDF)/VT
         GAMMAF = GAMMA
         AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C        PROPORTIONAL NAVIGATION IN AZIMUTH
C
         AYC    = LAMDAZ*VT*SIGDAF/G
         PHIC   = ATAN2(AYC,AZC)
         AZC    = AZC*COS(PHI)+AYC*SIN(PHI)
         GO TO 100
    9    PH1    = 1
C
C *** PULLUP TO ATTACK ALTITUDE
C     PROPORTIONAL NAVIGATION IN AZIMUTH
C
   30 IF(ALTUDE.GE.ALTATT) GO TO 39
C
C        COMMAND ATTACK ALTITUDE
C
         ALTC   = ALTATT
         ALTUDF = ALTUDE
         GAMMAC = KALT*(ALTC-ALTUDF)/VT
         GAMMAF = GAMMA
         AZC    = COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C        PROPORTIONAL NAVIGATION IN AZIMUTH
C
         AYC    = LAMDAZ*VT*SIGDAF/G
         AZC    = AZC*COS(PHI)+AYC*SIN(PHI)
         PHIC   = ATAN2(AYC,AZC)
         GO TO 100
   39    PH4    = 1
C
C *** ATTACK - BANK-TO-TURN OR SKID-TO-TURN OR BOTH
C     PROPORTIONAL NAVIGATION IN AZIMUTH AND ELEVATION
C
   40    ALTC   = 0.0
         GAMMAC = 0.0
         GAMMAF = GAMMA
         AYC    = LAMDAZ*VT*SIGDAF/G
         AZC    = LAMDEL*VT*SIGDEF/G+COS(GAMMAF)
```

```
C
C             FOR FINAL ATTACK, ENABLE NEGATIVE G COMMAND AND LATERAL
C             G COMMAND (NYC), OR LEAVE NYC = 0.0.
C
              NZC     = AZC*COS(PHI)+AYC*SIN(PHI)
              NYC     = 0.0
C ************* PATCH TO ENABLE SKID TO TURN (KTEST = 1)
      KTEST = 0
      IF(KTEST.EQ.1) GO TO 95
C ************* END
C
C             TURN TO PLACE THE NEAREST NZ VECTOR (+ OR-) IN THE DESIRED
C             DIRECTION.
C
              PHI     = ANGLE(PHI)
              PHIG    = ANGLE(ATAN2(AYC,AZC))
              IF (ABS(PHIG-PHI).LE.PI/2.) GO TO 50
                  PHIC  =ANGLE(PHIG+PI)
                  GO TO 60
   50         CONTINUE
                  PHIC  = PHIG
   60         CONTINUE
C
C             ENSURE THE MISSILE ROLLS IN THE SHORTEST DIRECTION TO PHIC.
C
              DELPHI = PHIC-PHI
              DPHIAB = ABS(DELPHI)
              IF(DPHIAB.LT.PI) GO TO 100
              IF(PHIC.GE.0.0) GO TO 90
              PHIC    = PHIC+2.0*PI
              GO TO 100
   90         PHIC    = PHIC-2.0*PI
              GO TO 100
C
C *** FOR SKID TO TURN ONLY, SET PHIC TO 0.
C
   95         PHIC    = 0.0
              NYC     = AYC*COS(PHI)-AZC*SIN(PHI)
C
  100 CONTINUE
C ***********************************************************************
C
C         NZ COMMAND LIMITED TO -2 & +4 G'S.
C
      NZC= LIMIT(-2.0, 4.0,NZC)
      NYC= LIMIT(-1.0,1.0,NYC)
C
      RETURN
```

```
      END
C
C                                                                 9-06-84
C ***********************************************************************
      FUNCTION ANGLE(THETA)
C ***********************************************************************
C     LIMITS ANGLES IN RADIANS TO THE RANGE +PI TO -PI.
C     *******************************************************************
C
      IMPLICIT REAL(A-Z)
C
      DATA PI /3.1415962/
C
   10 IF(THETA.LT.PI) GO TO 20
            THETA = THETA - 2*PI
            GO TO 10
   20 CONTINUE
C
   30 IF(THETA.GT.-PI) GO TO 40
            THETA = THETA + 2*PI
            GO TO 30
   40 CONTINUE
      ANGLE      = THETA
      RETURN
      END
```

```
C
C
C
C                            APPENDIX F
C
C                   LISTING FOR SUBROUTINE MISSN2
C
C
C
C                                                          9-07-84
C ******************************************************************
      SUBROUTINE MISSN2
C ******************************************************************
C     BALLISTIC  GUIDANCE SCHEME. ALLOWS VARIABLE POPUP IN CLOSE
C     WITH NO OFFSET TURN. AFTER POPUP, MISSILE ROLLS TO 90 DEGREES
C     ANGLE OF BANK AND USES LATERAL AND VERTICAL PROPORTIONAL NAV.
C
C     MAKES MISSION PHASE DECISIONS AND INVOKES THE DIFFERENT MODES
C     OF GUIDANCE AS REQUIRED. DELIVERS NZC AND PHIC TO THE AUTOPILOT
C     CONTROL LOOPS. NYC IS ASSUMED TO ALWAYS BE ZERO. NZC IS LIMITED
C     TO +4.0 AND -2.0 G'S.
C     ***************************************************************
C
      IMPLICIT REAL(A-Z)
      INTEGER PH1,PH2,PH3,PH4,I,J,K,N,NPTS,CPA,NOLT,PCOUNT,NFAZE
C
      COMMON /A/ TIME   ,FINTIM,DT    ,CPOT ,NCUT ,NPTS ,CPA   ,PCOUNT,
     *           G      ,T     ,RHO   ,PI   ,PII  ,MASS ,WT    ,S     ,
     *           IXX    ,IYY   ,IZZ   ,IXZ  ,IA   ,IB   ,IC    ,CS    ,
     *           ID     ,IE    ,IF    ,IG   ,IH   ,II   ,IJ    ,IK    ,
     *           CHORD2,CHORD ,SPAN2 ,SPAN ,NFAZE
C
      COMMON /C/ KPTCHR         ,KROLLR         ,KYAWRT         ,KBANK        ,
     *           KGAMMA         ,KALT           ,CGARML         ,CGARMN       ,
     *           RRTLIM         ,PLIM           ,KNY            ,KNZ          ,
     *           AILRON         ,STBLTR         ,RUDDER         ,
     *           BSERO          ,NZSERO         ,NYSERO
C
      COMMON /D/ ALFA           ,BETA           ,VT             ,HMDOT        ,
     *           L              ,V              ,H              ,
     *           PHI            ,GAMMA          ,THETA          ,SY           ,
     *           CD             ,CY             ,CL             ,CR           ,
     *           CM             ,CN             ,P              ,Q            ,
     *           R              ,ALFADT         ,BETADT         ,PDOT         ,
     *           QDOT           ,RDOT           ,NZ             ,ALTUDE       ,
     *           XM             ,YM             ,XMDOT          ,YMDOT
C
      COMMON /F/ PH1            ,PH2            ,PH3            ,PH4          ,
     *           OFFSET         ,ALTATT         ,SGDZPU         ,MISDST       ,
```

```
     *              LAMDAZ       ,LAMDEL       ,KNFAZ        ,KNFEL       ,
     *              NZC          , PHIC        ,GAMMAC       ,PCLIM       ,
     *              PC           ,QC           ,RC           ,RANGE       ,
     *              SIGAZ        ,SIGEL        ,SIGDAF       ,SIGDEF      ,
     *              SYT          ,THETAT       ,XT           ,YT          ,
     *              HT           ,NYC          ,POPRNG
C
      COMMON /G/    AOA          ,SIDESL       ,BANK         ,FLTPHC      ,
     *              BANKC        ,PITCH        ,ROLLRT       ,ROLRTC      ,
     *              PTCHRT       ,YAWRT        ,HEADNG       ,FLTPTH      ,
     *              HEADT        ,ELEVT        ,DSIGAZ       ,DSIGEL      ,
     *              DSGDAZ       ,DSGDEL       ,ERFBK        ,ERFRR       ,
     *              ERFAZ        ,ERFEL
C
C EXECUTABLE STMTS:
C*************************************************************************
C
C     SET POPUP RANGE AND A PARABOLIC ALTITUDE TRACK
C
      POPRNG  = 15000
      VH   =  SQRT(XMDOT**2+YMDOT**2)
      ALTATT  = HMDOT*RANGE/VH + (G/2.)*(RANGE/VT)**2 + 10.
C
C     MISSION PHASE LOGIC AND GUIDANCE COMMANDS
C
    1 IF(PH4.EQ.1) GO TO 40
      IF(PH1.EQ.1) GO TO 30
C
C *** INGRESS FROM INITIAL CONDITION TO POPUP MANEUVER
C
         IF(RANGE.LT.POPRNG) GO TO 9
C
C           ALTITUDE HOLD
C
            ALTC    =  50.0
            ALTUDF  =  ALTUDE
            GAMMAC  =  KALT*(ALTC-ALTUDF)/VT
            GAMMAF  =  GAMMA
            AZC     =  COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C           PROPORTIONAL NAVIGATION IN AZIMUTH
C
            AYC     =  LAMDAZ*VT*SIGDAF/G
            PHIC    =  ATAN2(AYC,AZC)
            NZC     =  AZC*COS(PHI)+AYC*SIN(PHI)
            GO TO 130
    9       PH1     =  1
C
```

```
C ***     PULLUP TO ATTACK ALTITUDE
C         PROPORTIONAL NAVIGATION IN AZIMUTH
C
C ************* PATCH  TO RETAIN BTT GUIDANCE TO IMPACT (KTEST=1)
        KTEST = 1
        IF(KTEST.EQ.1) GO TO 32
C ************* ENDPATCH
   30     IF(ALTUDE.GE.ALTATT) GO TO 39
C
   32     CONTINUE
C             COMMAND BALLISTIC ATTACK ALTITUDE
C
              ALTC   =  ALTATT
              ALTUDF =  ALTUDE
              GAMMAC =  KALT*(ALTC-ALTUDF)/VT
              GAMMAF =  GAMMA
              AZC    =  COS(GAMMAF)+KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C             PROPORTIONAL NAVIGATION IN AZIMUTH
C
              AYC    =  LAMDAZ*VT*SIGDAF/G
              NZC    =  AZC*COS(PHI)+AYC*SIN(PHI)
              PHIC   =  ATAN2(AYC,AZC)
              GO TO 100
   39         FF4    =  1
C
C ***     ATTACK
C         PROPORTIONAL NAVIGATION IN AZIMUTH AND ELEVATION
C
C             COMMAND ATTACK ALTITUDE & ROLL TO 90 DEG BANK.
C
   40         PHIC   =  PI/2.
              ALTC   =  ALTATT
              ALTUDF =  ALTUDE
              GAMMAC =  KALT*(ALTC-ALTUDF)/VT
              GAMMAF =  GAMMA
              AZC    =  KGAMMA*VT*(GAMMAC-GAMMAF)/G
C
C             PROPORTIONAL NAVIGATION IN AZIMUTH
C
              AYC    =  LAMDAZ*VT*SIGDAF/G
              NZC    =  AZC*COS(PHI)+AYC*SIN(PHI)
C             NYC    =  AYC*COS(PHI)-AZC*SIN(PHI)
              NYC    =  0.0
C
  100 CONTINUE
C ******************************************************************************
C
```

```
C
C      NZ COMMAND LIMITED TO -2 & +4 G'S; NYC TO +- 0.5 G.
C
       NZC= LIMIT(-2.0, 4.0,NZC)
       NYC= LIMIT(-0.5, 0.5,NYC)
C
       RETURN
       END
```

LIST OF REFERENCES

1. Patterson, Kent R., *Bank-To-Turn Cruise Missile Terminal Guidance and Control Law Comparison*, M.S. Thesis, Naval Postgraduate School, Monterey, California, 1983.

2. IBM Company, *Continuous System Modelling Program III (CSMP III) Program Reference Manual*, Program Number 5734-XS9, December 1975.

3. Speckhart, F.H., and Green, W. L., *A Guide to Using CSMP-The Continuous System Modeling Program*, Prentice-Hall, 1976.

4. Hewett, M.D., Guidance and Control Systems Course (Class Notes), 1983.

5. Roskam, J., *Airplane Flight Dynamics and Automatic Flight Controls*, Roskam Aviation and Engineering Corp., 1979.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Dr. Marle D. Hewett E.D Textron, Inc. 2485 McCabe Way Irvine, California 92714	2
4.	CDR Barton P. Anderson, USN Naval Air Rework Facility Naval Air Station Pensacola, Florida 32508	2

END

FILMED

5-85

DTIC